creative computing

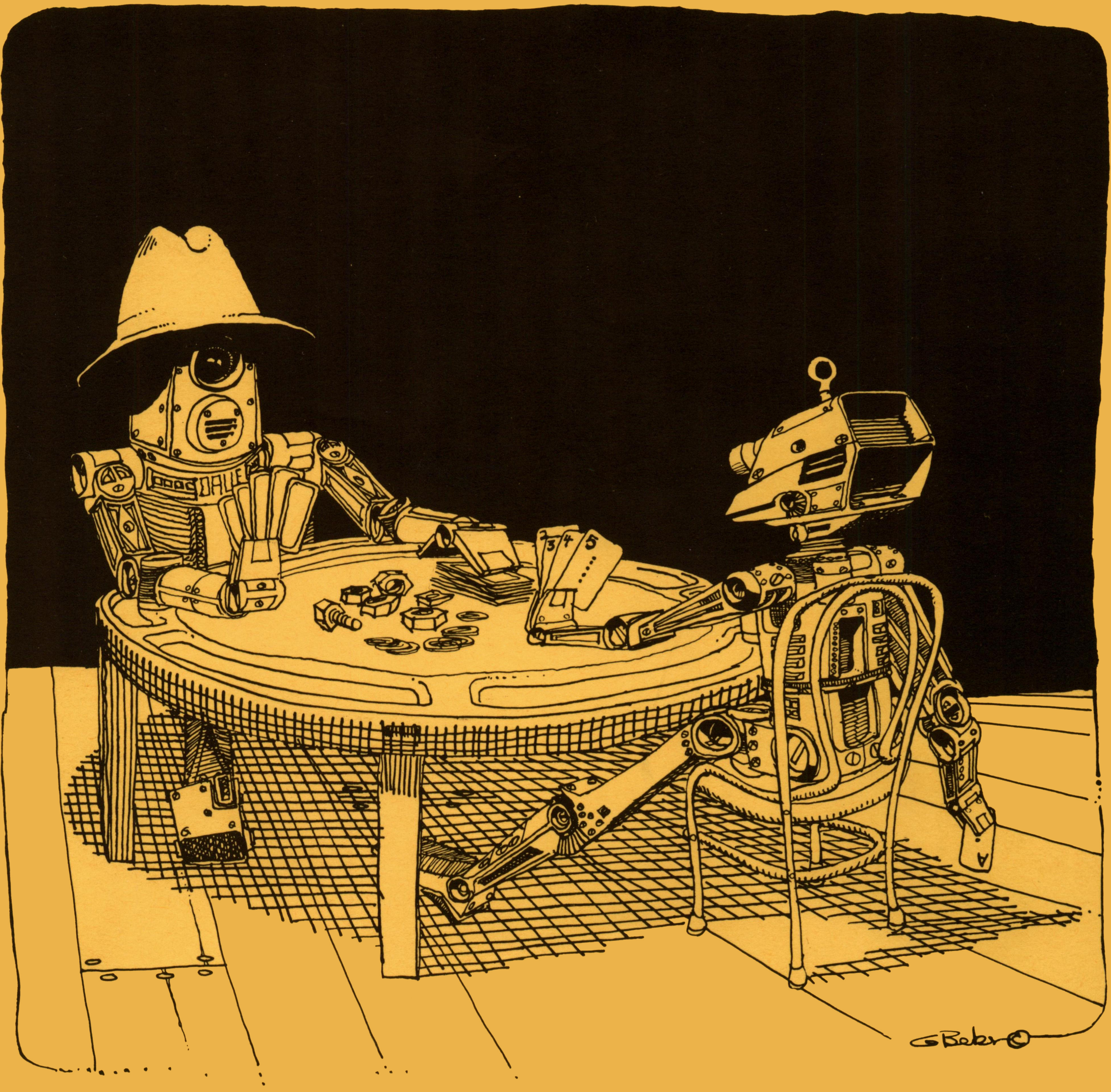
the magazine of recreational and educational computing

vol 2, no 1 Jan-Feb 1976 $1.50

GAMES & PUZZLES

☛ FOR POCKET CALCULATORS

☛ FOR COMPUTERS FOR HUMANS ☚



Pocket Calculator Tricks

GAMES: Roadrace, Chase, Dr. Z

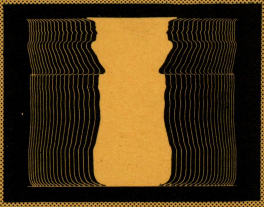
Learning With Computer Games

Creative Chess

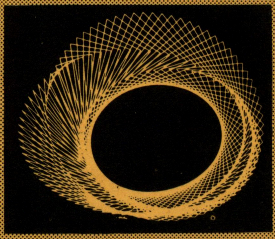
Magic Squares on the Computer

Computer Art for people

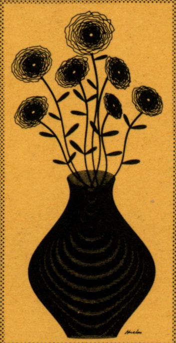
. . . the modern medium



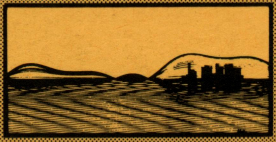
WOMAN 24x30
GOLD ON DARK GREEN
GOLD FRAME $42.50



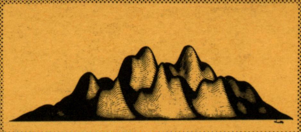
CRIMSON COSMOS 18x21
WHITE ON DARK RED
BLACK FRAME $32.50



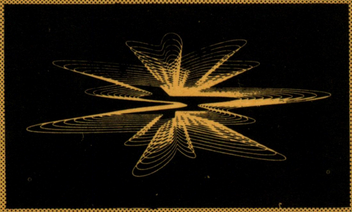
FLORES EN FORTRANES 16x25
RED, BLUE, GREEN
ON SKYBLUE
GOLD FRAME $38.50



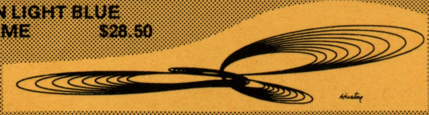
REFLECTIONS 13x21
GOLD ON DARK BLUE
GOLD FRAME $32.50



MOUNTAINS 13x21
GOLD ON DARK GREEN
GOLD FRAME $32.50



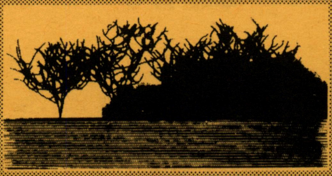
SONIC BOOM 13x21
WHITE ON LIGHT BLUE
PEWTER FRAME $28.50



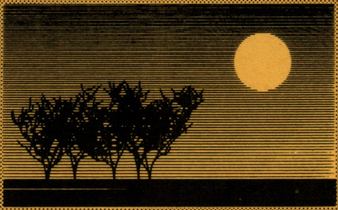
FREE FLIGHT 9x21
DARK BLUE ON SKYBLUE
PEWTER FRAME $28.50

LIMITED EDITION SERIGRAPHS

Katy Owens has created a beautiful and exciting work of Computer Art. She has authorized an edition of 300 signed and numbered serigraphs. The overall size is 30 by 40cm. The serigraph is done on yellow in sky blue, green, brown, and red. Set in the highest quality gold frame, glassed, mounted, and matted in rich brown, this work becomes quite a conversation piece.

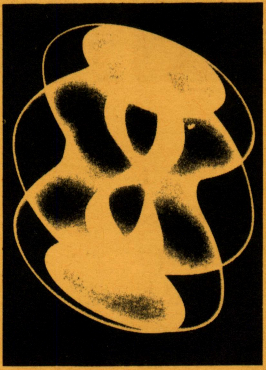


DAWN OF CREATIVE COMPUTING 12x16
RED, GREEN, BROWN AND
BLUE ON YELLOW
GOLD FRAME $24.00
Print only - $6.00

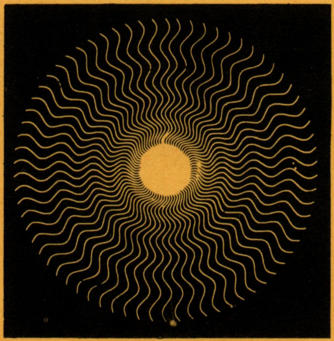


MOON RISE, another beautiful Owens serigraph on yellow in blue and black attractively matted, glassed and framed in pewter. $24.00 complete/$5.00 print only.

These attractive prints are drawn by a computer . . . carefully silk-screened . . . then custom-framed with the same high-quality, contemporary framing material used by custom framers and museums.



GRAFIKEN I 24x30
BLACK AND WHITE
SILVER FRAME $42.50



RA 21x21
FLUORESCENT ORANGE ON
BLACK
BLACK FRAME $32.50

All prints except Dawn and Moonrise are $2.50 ea.

COMPUTRA Inc. SUITE #2 3306 W. JACKSON MUNCIE, IN 47304

Dear Mr. Huston,
Please enter my order as specified below:

NAME ____________________

STREET ____________________

CITY __________ STATE __________ ZIP __________

Please charge my ☐ BANK AMERICARD ☐ MASTER CHARGE ☐ AMERICAN EXPRESS

Card Account No ____________________

Master Charger's bank code ______ Good Thru: Mo ____ Yr ______

Signature: ____________________

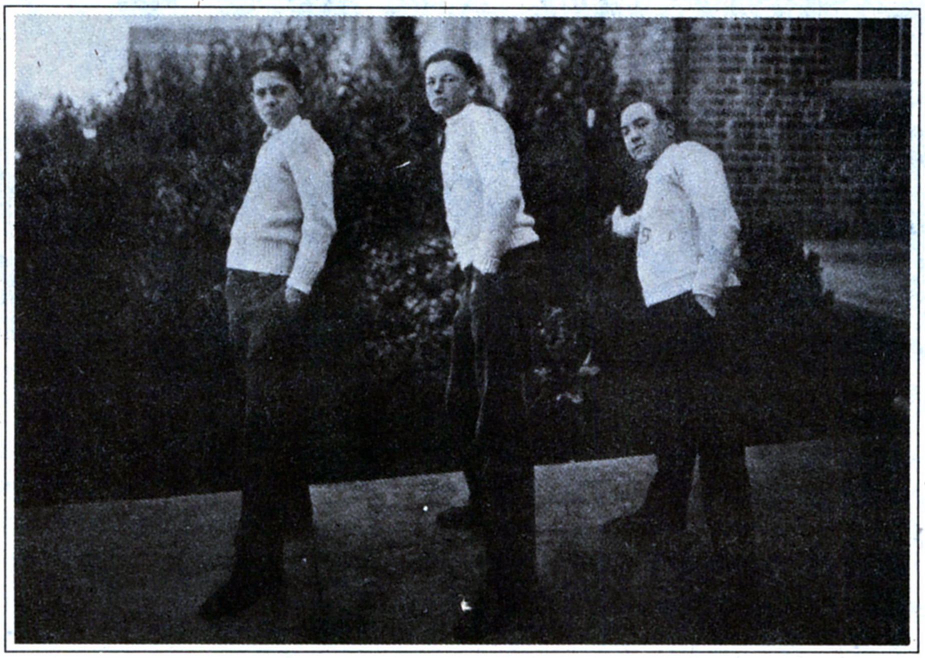
☐ Please find enclosed a check or money order.

QTY	TITLE	AMOUNT
	Subtotal	
	Shipping*	
	Sales tax (Indiana only)	
	TOTAL	

Both prints and framed pictures are carefully packed in heavy-duty containers to insure fast, safe delivery.

*Please add $1.00 shipping if ordering prints. When ordering framed pictures, please add $2.00 shipping PER PICTURE.

HEWLETT-PACKARD COMPUTERS. BECAUSE IT'S NOT RAH-RAH AND THE OLD SCHOOL TIE ANYMORE.



YELL LEADERS

The old, simple days of education are gone. Today's educational world is complicated and complex. It's full of records, paperwork and data which is exasperating to everyone involved.

That's why we decided to do something about it at Hewlett-Packard. We've worked hard at becoming the leader in designing computers for educational needs. Computers that can simplify and enhance your work, leaving you the critical time to bring about the educational programs you set out to accomplish. For instance, we've developed terminal-oriented computer systems and applications packages to meet instructional and administrative needs.

At Hewlett-Packard we can't bring back the good old days of education. But we can help you make the present educational process a lot more effective. Write Hewlett-Packard and find out all the facts. It could be the most important thing you've done for your educational system in a long time.

HP Computers.
They work for education.

HEWLETT hp PACKARD

Sales and service from 172 offices in 65 countries.
1501 Page Mill Road, Palo Alto, California 94304

☐ Send Elementary/Secondary School Literature.
☐ Send College/University Literature.
☐ Have an HP Sales Representative call.

Name
Title Dept.
Institution
Address Phone
City/State/Zip

22549

BASIC, COBOL, FORTRAN IV and assembler in one timesharing system.

$31,240

Yes, the price is printed correctly even though a computer system with these capabilities might be expected to cost ten times as much. ☐ The surprising power of this computer system comes from a remarkable new software operating system called ETOS (EDUCOMP's Timesharing Operating System) developed by Educomp Corporation. Using this system, a batch stream may be running from a card reader with output going to a line printer while *simultaneously* numerous other users may be running timesharing jobs from their individual terminals in BASIC, FORTRAN IV, COBOL, or Assembler Language. Or they may be using the system's powerful editor to create and modify data files. And, through the unique virtual memory technique employed by ETOS, each user may access up to 32K words of memory. ☐ While ETOS is sophisticated and powerful, it is also reliable and easy to use. Reliability is assured because ETOS is built around Digital Equipment Corp's dependable PDP-8 series computers and peripherals. Its ease of operation means that the system can run all day virtually unattended. ☐ At Educomp, we know educational computer systems. They're our only business. You're welcome to call on and talk to any one or all of our 100 plus satisfied customers. ☐ In addition to ETOS, we have single-user BASIC systems for under $5700 complete, timesharing systems, and an incredibly comprehensive general-purpose data retrieval system called GPRS. ☐ Please write or call with your educational computing requirements. You may specify your area of special interest or let us help you define your requirements.

educomp
corporation

196 Trumbull Street, Hartford, CT 06103, (203) 728-6777

Educomp Corporation
196 Trumbull Street
Hartford, CT 06103

Please send information on ☐ ETOS, ☐ GPRS,
☐ Educomp BASIC, ☐ COBOL,
☐ Complete Educational Computer Systems,

☐ Other____________________

Name______________________________

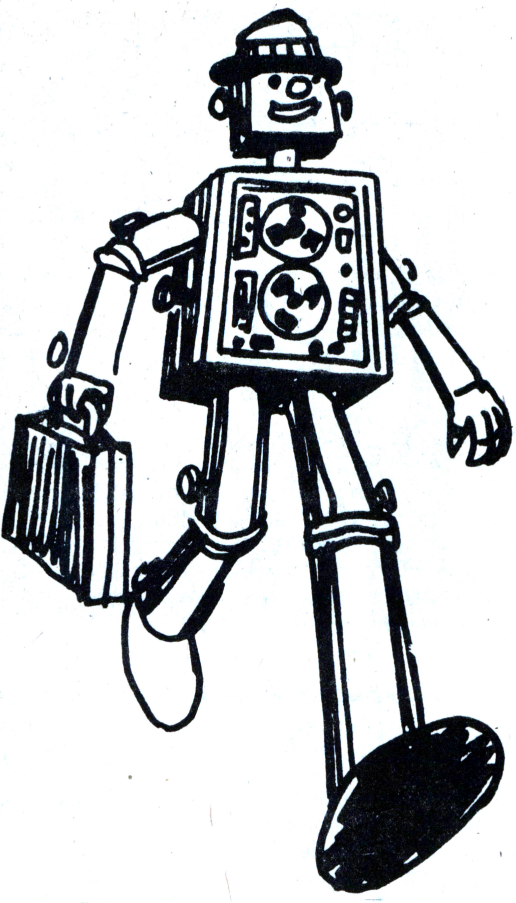
School______________________________

Address______________________________

City____________________ State________

Phone____________________ Zip________

creative computing



In This Issue of CREATIVE COMPUTING

SPECIAL GAMES & PUZZLES ISSUE

ARTICLES

Learning with Computer Games — *Ahl* 6
Building a MITS Altair 8800 — *Gray* 13
Calculators in the Classroom — *Pendleton* 16
Beating the Game — *Thomsen* 22
Computer - Planned Snowmen — *McLean* 24
Magic Squares on the Computer — *Piele* 28
The Computer Glass Box — *Peelle* 35
Small Computers Play Chess 40
About Computing — *Chase* 41
David vs. 12 Goliaths — *Newborn* 42
The Mystic 7 — *Dickins* 61
Non-usual Mathematics — *Reagan* 64
Simulated Strategies of Game - Playing — *Reisman* 65

THINGS TO DO

Pocket Calculator Tricks 18
7 Pocket Calculator Games — *Rogers* 19
The Keyboard Game — *Yarbrough* 20
The Tower of Brahma 25
Creative Chess — *Koetke* 39
Puzzles - 15 Big Pages — *Ahl* 45
Turning a Puzzle Into a Lesson — *Homer* 62

COMPUTER GAMES

Wumpus -2 (Wumpus is back!) — *Yob* 66
War 3 (Another gunner game) — *Forman* 69
Dr. Z (Computer psychiatrist) — *Zeddies* 69
Concentration (Like the card game) — *Calter* 72
Roadrace (Along NY Rt. 20) — *Cotter* 73
Condot (Connect the dots) — *Lund, Olivieri* 75
Chase (High-voltage maze) — *Cotter* 75

NEWS, NOTES, SOURCES

Compleat Computer Catalogue (Part 3) 11
Tips for Buying a Pocket Calculator — *Weaver* 17
General Purpose Calculator Ratings 18

FOOLISHNESS

Ode to a School Computer — *Ahl* 10
Span-O-Vision — *Johnson* 12
The Sleeping Queued T — *Ludwig* 26
Conceptual Chess — *Johnson* 43
The Theory of Relativity — *Payack* 44

DEPARTMENTS

Notices 4
Input/Output 9
Reviews 79
Review of Space 1999 83
Feature Review 84
(34 Books on BASIC)

Editor-in-chief and Publisher
David H. Ahl

Higher Education Editor
A. Kent Morton, Dartmouth College

Reviews Editor
Peter Kugel, Boston College

Problems Editor
Walter Koetke, Lexington High School

Careers Editors
Eleanor Corr, Keystone Jr. College, PA

Humor Editor
Jack Ludwig, King of Prussia, PA

Technology Editors
Tom R. Kibler, Atlanta, GA
Alan B. Salisbury, Eatontown, NJ

Associate Editors
Susan Hastings, Florham Park, NJ
Trish Todd, Brown University
John Lees, University of Missouri

Contributing Artist
George Beker, Stratford, CT

Business Staff
Carol Tick, Bernardsville, NJ

Volume 2 Number 1

CREATIVE COMPUTING is published bi-monthly by Ideametrics, (Editorial office) 20 Lynnfield Drive, Morristown, NJ 07960.

Subscriptions: Institutional 1-year $15, 3-year $40; Individual 1-year $8, 3-year $21; Student 1-year $6, 3-year $16. Foreign postage $1.50 per year additional. Single copy $1.50. Subscription orders, Form 3579, Change of address to CREATIVE COMPUTING, P.O. Box 789-M, Morristown, NJ 07960.

Copyright © 1975 by Ideametrics. All rights reserved. Reproduction prohibited. Printed in USA.

* * * Circulation Under 1,000,000 * * *

THE COVER

The cover illustration is by George Beker, a videotape editor, producer, and consultant, in Stratford, Ct.

Notices, etc.

FREE ART BOOK NOTICE

All subscribers who are current as of May-Jun 1976 will receive a $5.95 Computer Art book as part of their subscription. This fantastic book will be printed on high-quality coated paper, will have full-color illustrations and will portray the works of 40 leading computer artists from Germany to Japan to the USA. The book is edited by Ruth Leavitt, a computer artist (*Creative Computing* cover, Vol. 1, No. 2) residing in Minneapolis.

This book will replace both the May-Jun and Jul-Aug 1976 issues. We will resume in standard magazine format with the Sep-Oct 1976 issue.

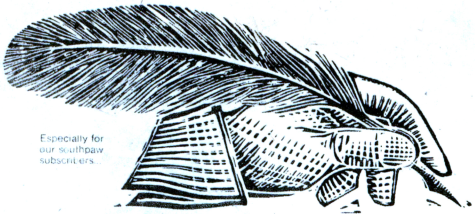
With luck (and persistence) the art book will be available in most bookstores across the U.S.; however, we will include direct ordering instructions in the Mar-Apr 1976 issue of *Creative* for those who want extra copies for gift giving, etc.

HAVING A MATH/COMPUTER/SCIENCE FAIR? WANT TO MAKE SOME MONEY?

You can sell copies of *Creative Computing* at your next math/computer/science/other fair, conference, or club meeting. Simply send us an order for bulk copies. Be sure to mention that it is for a school or non-profit organization, and you get a 50% discount off the $1.50 cover price. (Bookstores and commercial resalers get a 40% discount.) Please note: 1) Minimum order 25 copies, 2) Order at least 4 weeks in advance to allow enough shipping time, 3) Include your payment (75¢ x number of copies plus $1.00 for shipping). We can generally supply copies of the current and one previous issue in any mixture. At fairs where this has been done, they have sold quantities ranging between 32 and 290 copies (and the students have earned from $24 to $217).

THANKS

A special word of thanks to Graham Kimball of Carleton College, Northfield, MN for the loan of his 15-year collection of computer related cartoons. As soon as we iron out the permissions problems, you'll begin to see some of them on the pages of *Creative.*



Especially for our southpaw subscribers...

A MATTER OF ECONOMICS

Creative Computing has expanded a long way on subscription revenue alone. However, there comes a time in every magazine's life when it must seek advertising. In particular, each page of advertising allows us to run an additional 6 to 8 pages of editorial content (articles, games, stories, etc.). Five additional pages of advertising would permit us to run 100-page issues.

Unfortunately most potential advertisers say, "Why should I advertise with you? Who reads a magazine on (yeech) newsprint?" Frankly I don't have time to talk to every potential advertiser, so I'm asking you readers to write a note to the Marketing Manager or Advertising Manager of your favorite computer vendor/book publisher/electronics manufacturer/etc. and tell them they ought to advertise in Creative.. Thanks. – DHA

ONLY THE COMPUTER KNOWS FOR SURE

On your address label, you will find a "GC" followed by a number. To the direct mail service who keeps track of our subscribers, "GC" identifies *Creative Computing* and the number identifies each individual subscriber.

There is also a two-digit number on the label like 21 or 42. This indicates the Volume and Issue Number when your subscription expires, thus a 24 would indicate a subscription that expires with Vol 2, No 4. There are 6 issues per volume (or year).

It helps us immensely if you renew before your subscription expires, but *be sure to indicate on your order that it is a renewal.*

UNCOOL HAPPENINGS DEPARTMENT

Digital Equipment Corp. has withdrawn as a sponsor of the National Student Computer Fair.

On the happy side, however, Wang joined as a sponsor. For details about the Fair, See Creative Vol 1, No 5 (pg 32) or No 6 (pg 43).

QUANTITY PRICES FOR SCHOOLS, CLUBS, ETC.

Quantity	*1-Yr Bulk Subscriptions*	*Current Issue*	*Back Issues*
1-9	$8.00	$1.50	$1.50
10-24	7.00	1.00	1.25
25-49	6.00	.75	1.00
50-99	5.00		
100 +	4.00		

creative computing

BACK ISSUES

NOV-DEC 1974 – CHARTER ISSUE (Out of print)

JAN-FEB 1975 – GENERAL TOPICS (Out of print)

MAR-APR 1975 – COMPUTER CAREERS (Out of Print)
A Computer Career for You?
Profile of the Computer Industry – Job opportunities.
Shorthand Instruction via Computer.
Computer Cartoons – a complete analysis.
Games: Rabbit Chase, Not One, Binary.
Review of BASIC books – Summary and first four reviews.

MAY-JUN 1975 – SPACE EXPLORATION
Games: Super Star Trek, Lunar Landing, Parachute Jump on Any Planet, ICBM Missile Intercept, Seawar, Geowar.
Escape – A simulation about escaping the earth's gravitation.
Mr. Spock's 7th Sense – real computer sensors.
Trends in Mathematics Curriculum Research.
Computer Simulation of the Atmosphere.
Structured Programming.
The Automobile and Air Pollution – Writing your own computer simulation.
Mr. Spock centerfold poster.

SEP-OCT 1975 – COMPUTER LITERACY
Activities that don't require a computer.
Games: Guess, Wumpus, Civil War, Reverse, Schmoo.
Languages for learners: LOGO, Smalltalk.
Toward a Human Computer Language.
Building A Computer of Your Own – Part 1.
"Compleat Computer Catalogue" – (First installment)

NOV-DEC 1975 – COMPUTER IMPACT ON SOCIETY
Isaac Asimov: "Life & Times of Multivac"
Computer Threat to Society
Turning On With Computer Art
Technology . . . Doomsday for Individualism?
Survey of Public Attitudes Towards Computers

Back issues are $1.50 each paid in advance. A billing charge of $1.00 is added to all orders without payment enclosed.

MEMBER
COSMEP
COMMITTEE OF SMALL MAGAZINE
EDITORS AND PUBLISHERS

COMING SOON

MAR-APR 1976 – ARTIFICIAL INTELLIGENCE AND FUTURE COMPUTERS
Smaller, faster, cheaper computers.
Access to computers in museums, libraries and storefronts.
At what point does a computer start to think?
Review of every BASIC book in print (6th and last installment).

MAY-JUN 1976 – COMPUTER ART

This fantastic double-size, full color, quality paper, large format issue will feature the works of 40 leading computer artists from around the world. Guest editor for this special issue is Ruth Leavitt, an outstanding computer artist.

Closing date for articles or other material for issues is 3½ months before the cover date. If you want an acknowledgement of your submittal, please include a self-addressed stamped envelope.

BULK SUBSCRIPTIONS

Bulk subscriptions are now available to *Creative Computing*. All copies must be sent to the same address; minimum term of the subscription is one year.

Quantity	*Price*
10-24	$7.00 each
25-49	6.00
50-99	5.00
100+	4.00

CHANGE OF ADDRESS

Moving? Changing jobs? If so, to insure you get every issue of *Creative Computing*, please send us both your OLD and NEW address. Your "GC" code number which appears above your name on the address label would help too. And by all means, don't forget ZIP code. (We heard recently that the USPS automatically incinerates all magazines without zip codes.)

CREATIVE COMPUTING SUBSCRIPTION FORM

Published bi-monthly – 6 issues per year

All subscribers will receive a $5.95 computer art book FREE!

1. *Quan.*	*Type*	*Term*	*Price Each*
______	Institutional	1-Year	$15.00
______	Institutional	3-Year	$40.00
______	Individual	1-Year	$ 8.00
______	Individual	3-Year	$21.00
______	Student	1-Year	$ 6.00
______	Student	3-Year	$16.00
______	Bulk (10-24 copies)	1-Year	$ 7.00
______	Bulk (25-49 copies)	1-Year	$ 6.00
______	Bulk (50-99 copies)	1-Year	$ 5.00
______	Bulk (100+ copies)	1-Year	$ 4.00
______	Sample Copy		$ 1.00
______	Foreign Postage	$1.50 per year	

2. *New or Renewal*

☐ New
☐ Renewal
☐ Gift

3. *Billing Instructions*

☐ Payment enclosed
☐ Please bill me ($1.00 billing charge will be added except on institutional and bulk subscriptions.

4. *Name and Address*

Name ______________________ Title/Dept. __________

School or Company ______________________

Street Address ______________________

City ______________ State ________ Zip ________

Return form to: CREATIVE COMPUTING, P.O. Box 789-M, Morristown, New Jersey 07960

CREATIVE COMPUTING Editorial

Learning With Computer Games

Brief History of Sports and Games

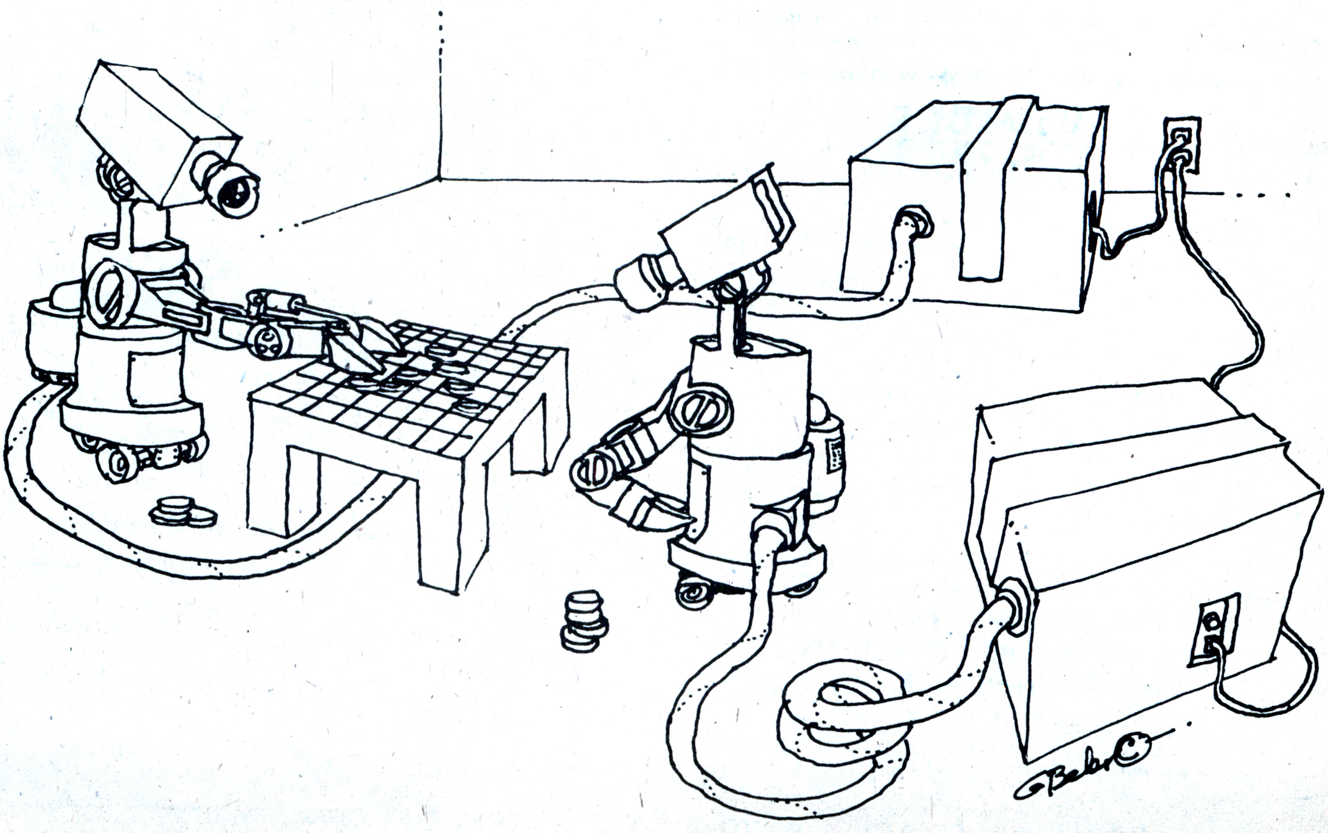
Sports and games have been with us almost since the dawn of man. The ancient Chinese and Egyptians devised an astonishing array of mathematical and logical games (Nim, Towers of Hanoi, Awari or Kalah, etc.). The Greeks originated a great many physical games in their Olympic competitions. Medieval Europe was responsible for many games played for recreation in nobles' courts (Chess, Hi-Q, etc.). Team games are a phenomenon of the last few centuries, with many originating in England and the United States.

Throughout history, the common thread running through all sports and games is that their intended function has been purely recreational. Games were viewed as a diversion from the realities of life. Secondly, sports and games were and are a leisure-time activity. In early Europe, only the nobles, who had leisure time, played games. Even today, except for professional athletes, sports and games are generally considered outside, extra-curricular activities.

Nevertheless, sports and games have generally had some redeeming virtues. Merely serving as a break from the realities of life is probably enough to assure games a place in history. But there is more. Chinese philosophers spoke of games as sharpening one's wit. Sports build the body just as card, board and mathematical games build the mind. Team games heighten the spirit of trust and cooperation.

Games as an Educational Tool

Not until the last 10 or 15 years of educational innovation, however, have games ever been used primarily as an educational tool where learning is the primary purpose of playing. Today, there is a growing body of evidence which indicates that a combination of learning games and student teams may well be one of the most effective approaches to learning ever devised. In a study by Edwards, DeVries, and Snyder (1972), the games-teams combination in seventh grade classes resulted in a significant increase on several dimensions of mathematics achievement. A follow-up study in 1973 found that both games and teams represent useful techniques for restructuring classroom processes. Their effects are complementary in that they create very different classroom experiences for students. Games cause students to translate their increased interpersonal interaction into increased informal peer-tutoring on the subject material at hand. They are also likely to view their class in a much more positive light. The addition of teams to a traditional classroom meets a rather different set of objectives. Attending the class and encountering the subject matter is not necessarily made more fun as it is with games. However, students work together on traditional classroom tasks to a much greater degree, resulting in an increased level of mutual concern among the students.



Other studies echo the results above. The learning effectiveness of games has been frequently cited (Allen, Allen, and Ross, 1970; Boocock and Schild, 1968; Fletcher, 1971). Learning games generally create an intense and often enjoyable interpersonal experience. This is due in part to the interdependent task structure which requires interaction among the players (Inbar, 1968).

Teams have been in use longer as an educational technique. Generally it has been found that students in a team outperform students working as individuals. A key reason for the effect of teams has been the high rate of peer tutoring (Wodarski, Hamblin, Buckholdt, and Ferritor, 1971).

As mentioned earlier, because games and teams positively affect different classroom processes, combining the two techniques creates an even more powerful effect on the learning environment. This in turn is translated into greater academic growth as well as increased trust and cooperation.

Computer Games and Simulations

People have written games for the computer almost since its birth. Many tended to be copies of real life sports and games (football, poker, tic-tac-toe, etc.). Other games were written as simulations of other real processes with a recreational quality (lunar landing, parachute jump, artillery weapon firing, etc.). Still other computer games were devised mainly for their recreational and mind-challenging value (game of life, bulls and cows, bagels, mugwump, etc.).

The educational value of both playing and writing computer games is substantial. They make ideal supplemental learning experiences when teaching about probability, statistics, logic, problem solving, decision making, and value clarification.

David H. Ahl

BIBLIOGRAPHY

Allen, L. E., Allen, R. W., & Ross, J. "The Virtues of Nonsimulation Games." *Simulation and Games,* 1970, *1,* 319-326.

Boocock, S. S. & Schild, E. O. (Eds.) *Simulation Games in Learning.* Beverly Hills, CA: Sage, 1968.

DeVries, D. S. & Edwards, K. J. "Learning Games and Student Teams: Their Effects on Classroom Process." *American Education Research Journal,* 1973, *10,* 307-318.

Edwards, K. J., DeVries, D. L., & Snyder, J. P. "Games and Teams: A Winning Combination." *Simulation and Games,* 1972, *3,* 247-269.

Fletcher, J. L. "The Effectiveness of Simulation Games as Learning Environments." *Simulation and Games,* 1971, *2,* 425-454.

Inbar, M. "Individual and Group Effects on Enjoyment and Learning in a Game Simulating a Community Disaster." In S. S. Boocock & E. O. Schild (Eds.), *Simulation Games in Learning,* Beverly Hills, CA: Sage, 1968, Pp. 169-190.

Wodarski, J. S., Hamblin, R. L., Buckholdt, D. & Ferritor, D. "Effects of Individual and Group Contingencies On Arithmetic Performance." Paper presented at the meeting of the American Educational Association, New York, January, 1971.

DEFINITIONS

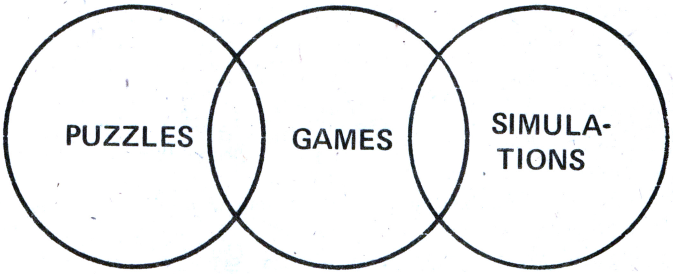
In the world of computer education and recreation a number of terms are used which are sometimes confused with one another.

A *Puzzle* is a problem that has a baffling quality or great intricacy, that one must exercise substantial mental ingenuity and thought to solve.

A *Game* is an amusement or sport involving competition under a specific set of rules. The competition may be against oneself or against laws of chance such as in solitaire or roulette.

A *Simulation* is a model or representation of a real-life process, situation, or event, frequently on a computer.

One might think of puzzles, games and simulations as having a certain overlap as follows:

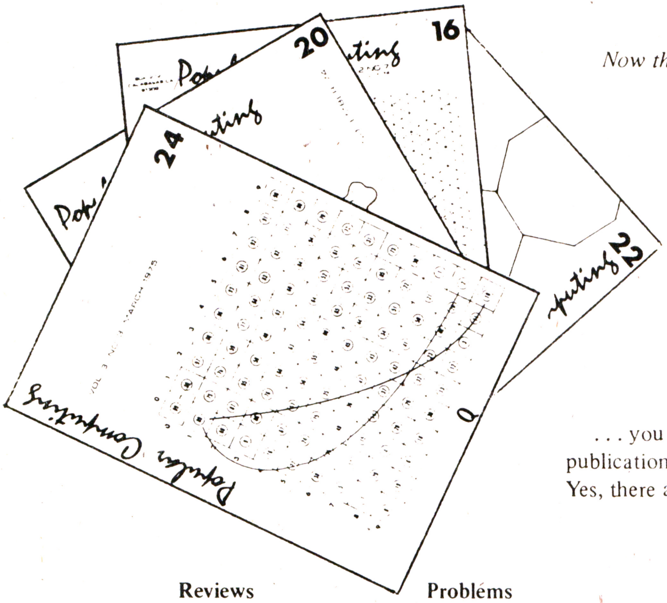


This Publication is Available in MICROFORM from...

Xerox University Microfilms

300 North Zeeb Road,
Ann Arbor, Michigan 48106
(313) 761-4700

PLEASE WRITE FOR COMPLETE INFORMATION



Now that you have it all together, you're ready for

Popular Computing

Now that you have the theory under your belt . . .

Now that you have assembled your own mini . . .

Now that you're ready to compute . . .

. . . you should subscribe to POPULAR COMPUTING, the only monthly publication designed for computists (those interested in the art of computing). Yes, there are other people like you who are interested in the art of computing.

FEATURES

Reviews	**Problems**	**Reader Interaction**	**Facts**
Books on the art of computing Electronic desk calculators	For students and teachers Flowcharts, and solutions	Problem solutions Comments and results	Timeless information, so the value of your copies will not decrease with time. Facts about certain integers (including roots and logarithms) to very high precision.

All this, plus the only on-going series of essays on the Art of Computing.

BOX 272 CALABASAS, CA. 91302

One year (U.S.) $15 when remittance accompanies order.

. . . The art of thinking one is thinking is best comprehended with

THE JOURNAL OF

Irreproducible Results

Box 234, Chicago Hgts., Illinois
60411 - U.S.A.

Please reserve current volume (4 issues per year) at a cost of $3.50 per year. Foreign Subscriptions $4,25.

Name ____________________

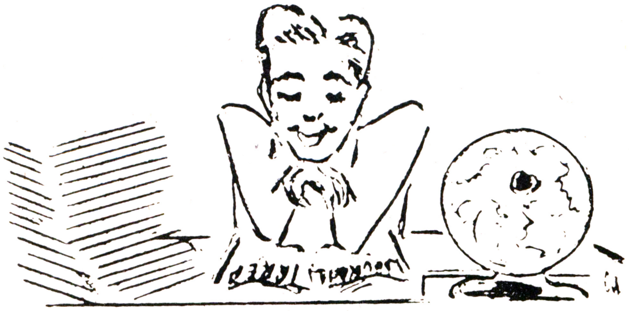
Address ____________________

City State Zip
(If gift subscription —)

From: ____________________

Enclosed is check ☐ cash ☐ money order ☐

SEND A GIFT SUBSCRIPTION TO THE HEAD OF YOUR DEPARTMENT OR SOME OTHER FRIEND



A Journal Of Satire And Humor Read From Cover To Cover

Input/Output

Dear Editor:

A few comments on Sept-Oct issue: Some nice, spirited Input-Output! A healthy sign, I trust. The "Compleat Computer Catalogue" is a good idea. "On Computer Languages" and "Toward a Human Computer Language" are a good start into the subject. On page 40 I found the Logo example rather confusing, though.

I liked "Hunt the Wumpus" but, just out of curiosity, why must Gregory Yob *shoot* the Wumpus? A Wumpus seems like a nice sort of creature and probably an endangered species, to boot. Why not *tag* the Wumpus (for a wildlife field survey), or photograph the Wumpus, or simply *find* the poor thing and see if it knows anything interesting? "Schmoo", now, is step in the right direction. Throwing mud I can see as a game. I would prefer that shooting things not be so universally associated with playing games.

"What do you value" and "Life Auction" are good exercises. It's certainly a good idea to start thinking about such things as young as possible.

"Learning about Smalltalk" I thought was really good. Keep printing things like that

"Trots and Bonnie" should be safe enough, but I suppose some people will be upset by "pooper scooper." Sigh.

In other words—a good issue.

I'm working on a short article on the various proposed interstellar messages. Are the decoding of the ones proposed and sent so far really *obvious* to a non-Earth being? I'll include a BASIC program for "decoding" messages of the prime factor bit string type if I can write a reasonable one, otherwise I'll do it in FORTRAN.

How would you feel about an article, or series of articles, on logic? Starting out at ground zero with syllogisms and working slowly up through propositional calculus and simple digital logic to simple first order predicate. Programming possibilities included, of course.

John Lees
Rolla, MO

What do other readers think about a series of articles on logic?—DHA

(MORE LETTERS ON PG. 34)

MORE STARTREK!

Dear Editor:

I really like *Creative Computing.* I think it is far superior to most of the other magazines that are available (and not trade journals like Modern Data). I am not a game freak (I do not even play my own STAR TREK game)—my interest is in simulation. STAR TREK acts as a good vehicle for simulating a "space war" environment. My goal is to make the game as challenging as possible by requiring good decisions by the players and by presenting a dynamic environment. Along these lines, your article about Forrester's "World Dynamics" fascinated me. I would like to see more about the Club of Rome reports in *Creative Computing.*

Most of the letters I have received from people concerning STAR TREK have been requests for copies of the program. Not too many have been interested in algorithm development. The latter was what I really intended. I.E.—a clearinghouse of "ideas"—not copies of various programs. In that context, I am disappointed in the response.

Enclosed is my version of STAR TREK (written in FORTRAN IV-G) that runs on our IBM 360-65. I don't know if you really want it—but here it is anyway.

Peter Weiss, COMSAT,
950 L'Enfant Plaza, SW,
Washington, DC 20024

Why not write Peter if you're seriously involved and interested in the development of new routines and algorithms for STAR TREK. We're not going to print the FORTRAN version in Creative *(it would run 12 pages). If you want a copy, write to Peter and, although he hasn't asked for it, why not send him $1.00 to cover his copying and mailing costs. —DHA*

HOORAY FOR BASIC!

Dear Editor:

Your magazine is great. It's just what us high school students who run BASIC on minis with TTY want.

Of course, BASIC has drawbacks and I wouldn't want to be confined to it for usage, but for understanding a program by reading it, BASIC is the best commonly used language. Reading, after all, is what the programs are for. Certainly, I like to run some of them but I don't have time to run them all, even though I sometimes pick new programming techniques from reading them.

When I do run them I usually make several revisions anyway and BASIC is easy to change into other languages, more so than the reverse. I don't feel you accomplish much copying a program exactly. Getting back to my original point, it is easier for the author to change to BASIC than for all the readers to try and decipher a more difficult language that they have no training in (if they even bother).

I am interested in learning new languages and I like your column "On Languages" and the articles about SMALL TALK and a Human Computer Language, but keep these programs in BASIC.

Paul McCullough
Flint, Michigan

YOUNGEST SUBSCRIBER?

Dear People:

Please send me a subscription to Creative Computing. I am 7 years old and am in the second grade at Earle C. Davis School.

Thank you.

Lynda Marie Wasula
North East, PA

Ode to a School Computer

by David Ahl

I was stayin' after school a week or so ago
'Cause I told a teacher where she could go

She had me settin' in this big old room
With a bunch of machines that just looked like doom

There's this big Mutha machine with flashin' lights
And a couple of funny-lookin' electric typewrites

Well I thought I'd type somethin' for the fun of it
So I hunted and pecked out just one word — "shit"

Before I could lean back in my chair and get steady
That machine typed WHAT, and then it said READY

So I typed a whole line of them four-letter words
But it just replied WHAT and READY like it hadn't heard

Well I figured since I couldn't go out fishin'
I'd teach that stupid machine to listen

So I picked up this book called *Teach Yourself BASIC*
And sat down at that Teletype prepared to face it

First I found to make that machine type my bit
I just had to put a PRINT in front of it

And then I found out that thing could add
And subtract and multiply and divide like mad

I found out too it knew all kinds of games
Like craps and blackjack and a cannon to aim

I was havin' all kinds of fun when that teacher walked in
She just looked at my output and started to grin

I kind of sheepishly asked if I could stay a while more
She said, "Sure; when you go just shut the door."

I tried some more games like football and poker
And a parachute jump written by some kind of joker

There was one where I could try to land on the moon
It would crash and blow up if I fired the engines too soon

Well, I played on through supper and into the night
And then finally quit when I saw dawn's first light

Some girls I know are a whole lot cuter
But I found a new kind of high with that computer

COMPLEAT COMPUTER CATALOGUE

RESOURCES
AND
TOOLS

We welcome entries from readers for the "Compleat Computer Catalogue" on any item related, even distantly, to computers. Please include the name of the item, a brief evaluative description, price, and complete source data. If it is an item you obtained over one year ago, please check with the source to make sure it is still available at the quoted price.

Send contributions to "The Compleat Computer Catalogue," *Creative Computing*, P.O. Box 789-M, Morristown, NJ 07960.

BOOKS AND BOOKLETS

TICCIT, PLATO IV REPORTS

Two evaluative progress/status reports about the PLATO IV and TICCIT CAI systems are available from the National Science Foundation. Some 800 PLATO IV plasma/microfiche terminals are now in use served by two host installations in Illinois and Florida. TICCIT uses a computer/user controlled color TV terminal and installations are operative in Utah, Arizona, and Virginia.

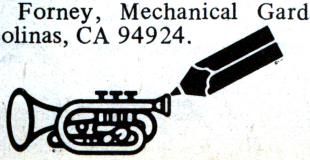
Also available from the NSF are summary abstracts of 1975 awards granted by the Technological Innovation in Education Section. Progress reports and awards listing free.

Erik D. McWilliams, Technological Innovation in Education, Directorate for Science Education, National Science Foundation, Washington, DC 20550.

MECHANICAL GARDEN

This wild 192-page book by Darrell Forney is a collection of graphics from the commonplace (old postcards, catalog illustrations) to the unexpected (schematic diagram of a hollow planet, UFO photo) to the bizarre (1890 African superstitions illustrated, sewing machine powered by a dog). Lots of fun! Printed on newsprint (like *Creative Computing*). $5.00 postpaid.

Darrell Forney, Mechanical Garden, Box 35, Bolinas, CA 94924.

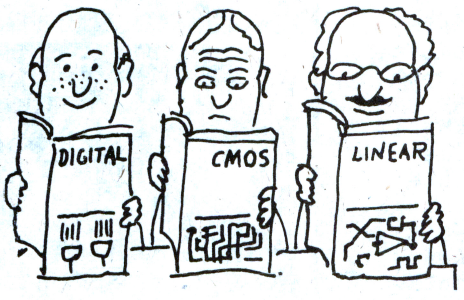


MAGAZINES, JOURNALS, NEWSLETTERS

COMPUTER NOTES

Computer Notes is a substantial monthly tabloid newspaper serving the Altair Users Group plus anyone else interested in the Altair line of computer kits. It carries articles by MITS engineers and software people as well as letters from users, questions and answers, software notes, etc. Dave Bunnell, the capable editor of *Computer Notes* usually has some interesting ramblings also – a description of the World's First Computer Store or the computer fair at Lawrence Hall of Science. WARNING: if you get the newsletter, you're probably going to want a computer kit very, very badly. *Computer Notes* is $10/year, sample copy free.

MITS, 6328 Linn N.E., Albuquerque, NM 87108.



BYTE

Byte is a new magazine edited by Carl Helmers who previously produced the *Experimenters Computer System* newsletter on building a computer from scratch. It bills itself as "the small systems journal" and basically covers the computer hobbyist field: hardware projects, using surplus equipment, tutorials, games, programming (in low-level or machine code), etc. The third issue (Nov 1975) contents are typical: ins and outs of volatile memories, chip designs, parallel output interfaces, a new ROM programmer, writing pseudo instruction sets, and the hexapawn game for machine code. For those into hardware, this is a great mag. Monthly. 96 pp per issue. $12/year, single copy $1.50.

Green Publishing, Peterborough, NH 03458.

THE COMPUTER HOBBYIST

An excellent technical/hardware newsletter. Tutorial articles, construction, cassette interface, surplus parts and how to use them, an 8008 graphics system, etc. Builders from scratch (of which I'm not one) seem to like this one the best. Monthly. $6/year. Back issues (started Dec 74) or sample copy 50¢.

The Computer Hobbyist, Box 295, Cary, NC 27511.

GAMES AND SIMULATIONS

SERIOUS COMPUTER GAMES

Eight serious computer games are available on paper tapes from ERC. They are: Planet Management, Mouse in the Maze, Stretching Springs, Oceanography, Friction Force, Paramecium Population, Photosynthesis, and Reaction of Magnesium in Hydrochloric Acid. $8.00 each.

Elaine G. King, Education Research Council of America, Rockefeller Building, Cleveland, OH 44113.

GAMES FOR CLASSROOM EARTH

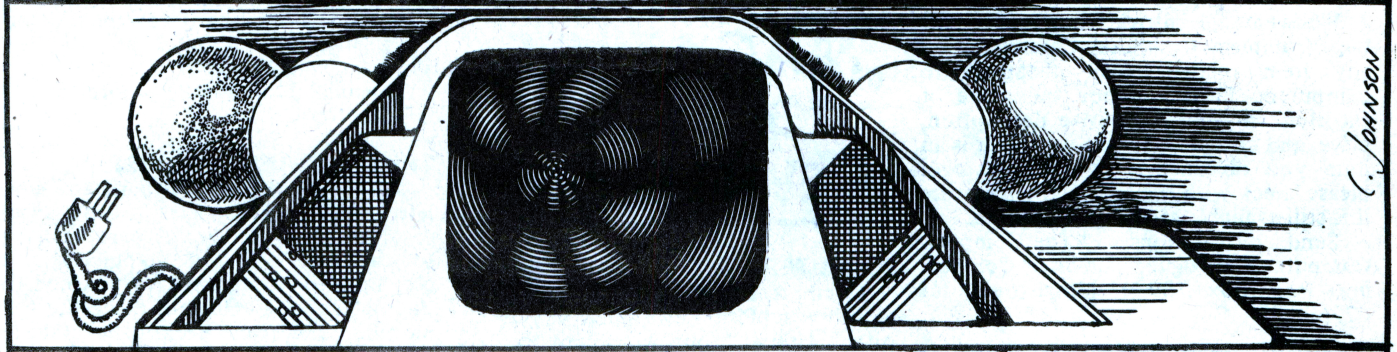
The idea game, fence game, freedom games, ethics game, gasoline game, an outdoor community games tournament, yes and even some computer games (teaser, king, and lunar landing) and many others are all in Zephyros Primer #14. Ron Jones, editor of this fabulous series of 15 primers, adds 3 or 4 new ones to the collection every year. Great graphics, intriguing new ideas. Primer #14 $2.50, others $1.50 to $4.50.

Ron Jones, Zephyros, 1201 Stanyan St., San Francisco, CA 94117.

Only 1 "Catalogue" page this month because we had so much other stuff!! DHA

A CREATIVE COMPUTING EXCLUSIVE!
SPANOVISION
No.2

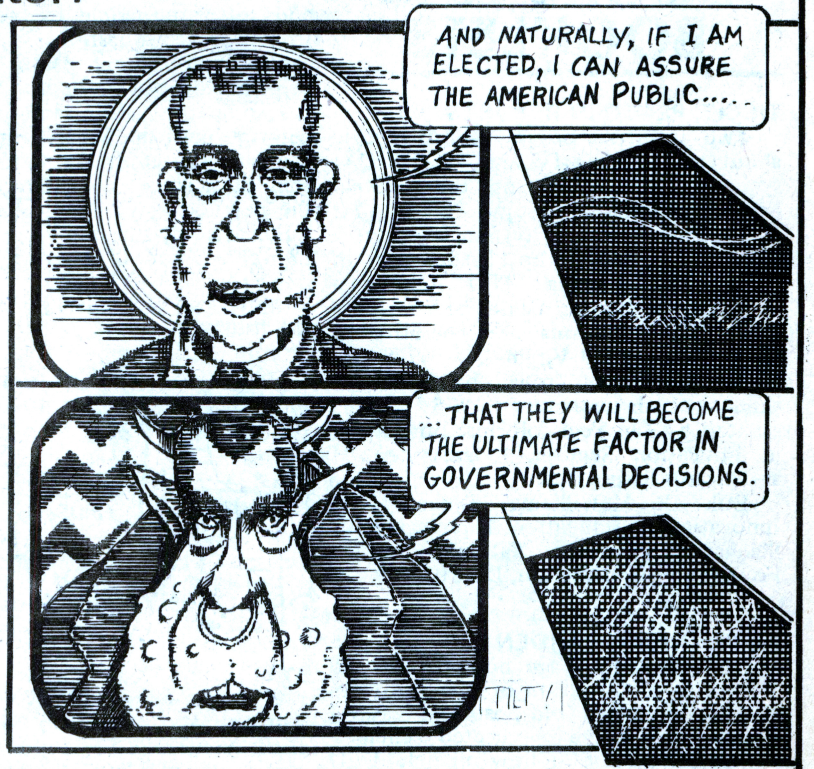
What suspense! Last issue, the massive machinations of Span-O-Vision produced its first VIS-U-COMP rendition, conjuring an unnerving scenario in which computers are nearly rejected en masse by an enraged society, only to rescue themselves with an astounding shift of focus! From the irritation of much misuse and quackery, commercial computer technology developes a pearl:



The Val-U-Graph Generator!

This device enables a home viewer to perceive when someone on T.V. is telling the truth! A special PSYCHOLOGICAL STRESS EVALUATOR UNIT reads variences in voice frequencies, determines stress, computes, and signals when a person is fibbing! Feb. 1987- Panicked legislators rush a bill through Congress prohibiting its use in legal proceedings and bans direct judgement of utterances on T.V..

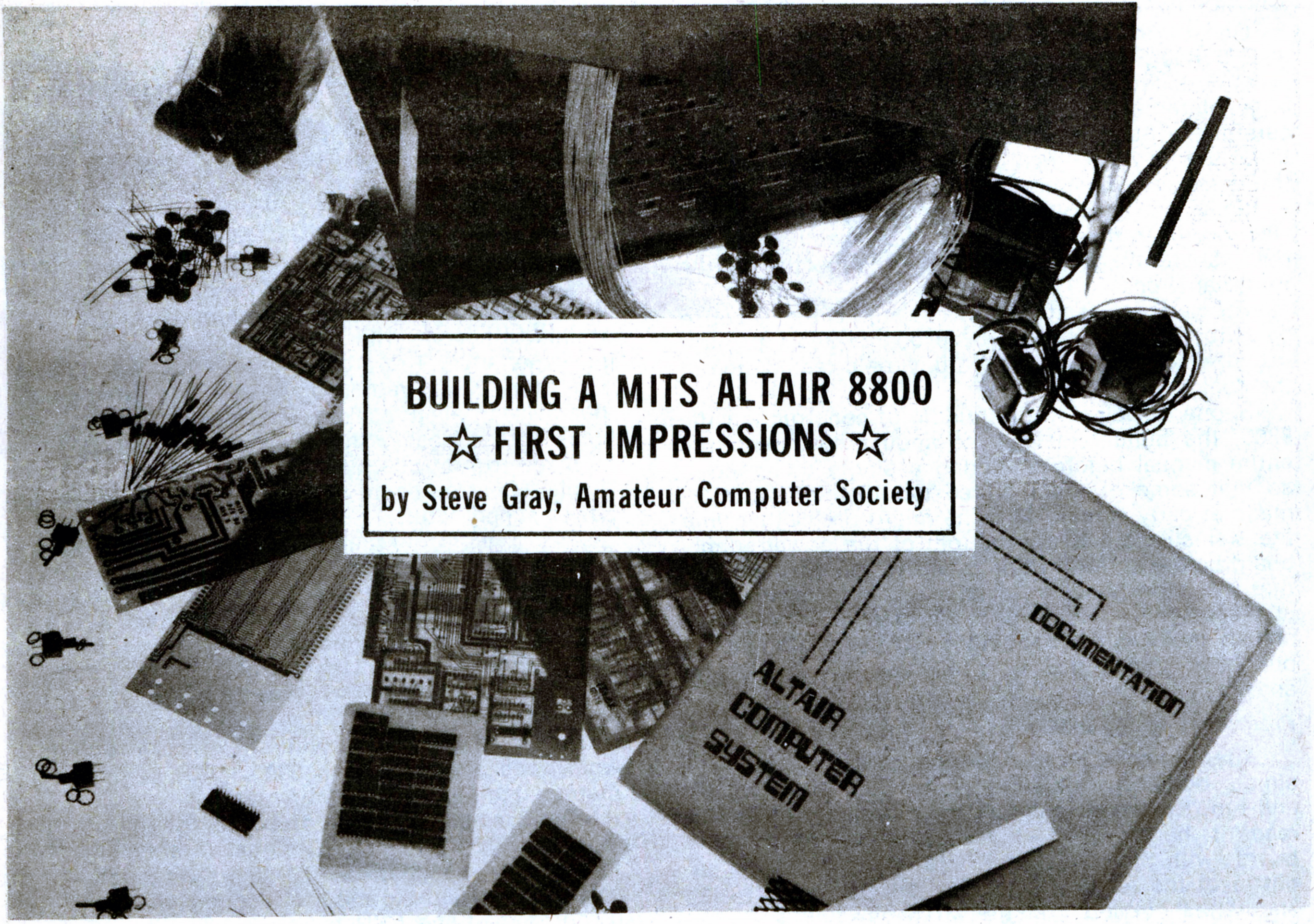
Undaunted, commercial interests and T.V. industry backers devise a prime-time series in which politicians are invited to debate. Computers are programmed with all pertinent information. The debator's voices are fed into the PS Evaluator, and converted into graphic images, colors, and simultaneous electronic soundtrack based on voice frequency modulations!


AND NATURALLY, IF I AM ELECTED, I CAN ASSURE THE AMERICAN PUBLIC....
...THAT THEY WILL BECOME THE ULTIMATE FACTOR IN GOVERNMENTAL DECISIONS.
TILT!

It is then the task of viewers to debate the credibility of the speakers from evidence within those images (by predetermined criteria)

Gosh! But amazingly, there's more. The home viewer may then extract claims of politician, punch into home terminal linked to huge atomic-powered ENCYCLOTRON (a SPAN-O-VISION evolvement) and receive a brief cinematic scenario immediately following said claim, along with its probability rating!

Once again the world is safe for democracy.



BUILDING A MITS ALTAIR 8800
☆ FIRST IMPRESSIONS ☆
by Steve Gray, Amateur Computer Society

Knocking the biggest is an old tradition. Today, IBM not only makes as much profit as all the other computer manufacturers combined, it also receives as much criticism as all the others put together. Some of the judgments are deserved. Still, IBM must be doing something right. Even though IBM computers aren't the fastest or the best, the company has built up a reputation for service and support that competitors find hard to beat. Detractors say that computer buyers are preconditioned by IBM's past record to accept IBM as the greatest.

MITS has sold thousands of the Altair 8800 microcomputer kits, many more than all the other kitmakers combined, and has been widely criticized in the hobbyist newsletters for late delivery, for delivery "in bits and pieces," for poor design, and for an inadequate manual. All true, yet the Altair continues to outsell all other kits. How long this will last, however, is a big question. Although MITS may be said to have gotten there "firstest," new companies are appearing with kits that may turn out to be the "mostest," and which may become bestsellers. But no matter how good a kit a competitor may have, he'll have to go a long way to make it as well-known as the Altair, which is heavily advertised, especially in technical trade magzines, several of which are bound—in the complete four-color, 24-page 8800 brochure.

Building the Altair 8800

Starting to build the 8800 is pretty much like starting a Heathkit radio receiver. There are instructions telling how, and a drawing that shows where, to install the ICs (integrated circuits), followed by similar sections on installing the resistors, capacitors, etc. The MITS manual is not as detailed as a Heathkit manual, nor would one expect it to be. However, there are some problems in building the 8800, sometimes due to information lacking in the manual, sometimes due to what seems to be poor mechanical design, and sometimes because of a lack of thoroughness by the manufacturer in checking out the parts lists, the parts supplied, and having the two match up. For instance, some of the solder-pads for capacitors are so close to other pads, or to printed wiring, that a solder bridge is all too easy to make. The installation of the heat sinks calls for silicone grease, yet the parts list doesn't mention any. So you go buy some, and after finishing the kit, and checking out the parts left over, you find that what looks like a bit of white paint, in a small capsule, is the heat-sink grease you needed. There are other leftover parts that will make the builder wonder if he left anything out. Each front-panel switch comes with two nuts, lock washer, and positioning ring. Yet the drawings show only a nut used with each switch. What about those other parts?

Anyone building an 8800 should first learn to tell the difference between 8-32 and 6-32 screws, and between #4 and #6 washers. Even after he does, he'll have trouble with the screws, nuts and washers, because what is supplied doesn't match up with what's called for.

Questions will arise that most builders won't be able to answer for themselves, and which they will probably have to write (or call) MITS about. For instance, "Is there any difference, on the 4K memory boards, between the SN74LO4 ICs called for, and the SN7404 ICs provided?"

The 66 Wires

One of the biggest problems, familiar to every 8800 builder, is that of the 66 three-foot wires, which must be soldered, one by one, into holes around the edge of the Display/Control board. No matter how careful the builder is, several are bound to pop off; if not at this point, then later when the other ends of the wires are soldered to the expander board. MITS eliminated this problem in their new kit for the Altair 680 (based on the Motorola 6800), in which the computer board plugs directly into a connector on the front-panel board.

The various resistors and capacitors that have to be outboarded on several 8800 boards seem to indicate that this Altair wasn't fully field-tested before the design was frozen.

Sooner or later, in the middle of constructing the 8800, the builder realizes he should have read the entire manual before soldering a single component, so that some of the required modifications can be made at early stages. Thus he might prefer to solder the ¼-inch braid to the bottom of the power-supply board before screwing this board to the chassis. Not that it's impossible to do later on, but it's not easy at the point it comes up in the manual, when there are so many wires hanging on it, and it must be put back into place with all those spacers and screws.

Programming the 8800

After finishing the 8800, the builder is given a simple addition problem to enter into the memory and run. But after he enters the program, and proofreads it, he may find—if his 8800 uses 4K memory boards with Signetic 2604 RAMs, many of which "do not meet the required specifications for access time and refresh period," as the MITS newsletter puts it (in recalling all such memory boards)—that various bits will drop out, and the program will not run correctly. This is somewhat disheartening, even if he can make the program work by relocating it in an unaffected part of memory. Only one other program is provided in the manual, a multiply program.

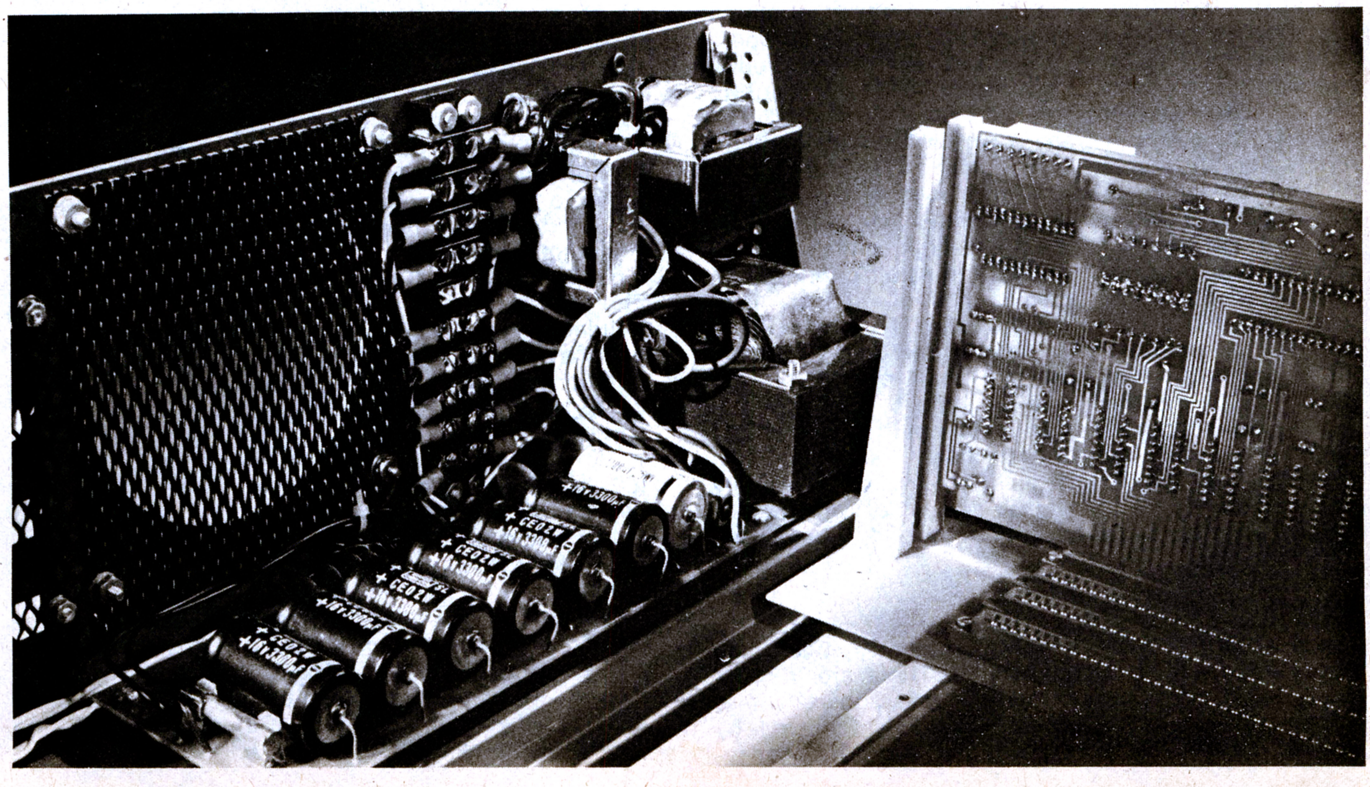
Checking out the finished 8800 requires a great deal of flipping the front-panel switches up and down. The switch-handles are so short, protruding less than 3/8 inch beyond the panel, that the front-panel lettering can soon be worn off, and one's nerves soon worn down as well. Longer handles, preferably with flattened ends, should have been used.

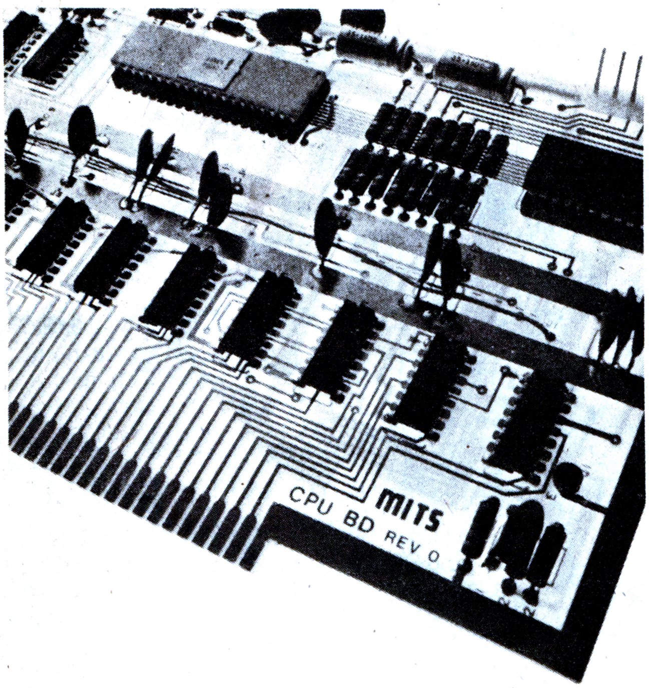
The 8800 Manuals

Let's take a closer look at the manual, or as MITS puts it, the set of manuals. The 82-page Assembly Manual is fairly well done, and shouldn't give much trouble to anyone with even a minimum of kit-building experience. The problems are more with assembling the parts, rather than with the instructions themselves, such as the 66 three-foot wires that break off so easily, the LEDs that somehow must be made to protrude just the right distance from the Display/Control board, the jungle of wires in the power supply, etc.

The "Theory of Operation Manual & Schematics" more closely resembles the set of notes a design engineer might make, and then hand to a technical writer with the request to "make a manual out of this." Only a computer-design engineer, or a hot-shot technician, could get a good understanding of how the 8080 works, from these 16 pages of text and a half-dozen schematics. The most important ICs are shown as blank boxes, without any explanation of what goes on inside: the 8080 MPU, 8212 (system status latch), and the memory ICs.

The discussion takes much for granted: "Note that the PRDY and PHOLD signals are synchronized to the leading edge of the 02 clock," although it isn't at all that obvious to most of the readers. The entire discussion of the CPU is on two pages, and that's all the Display/Control board operation gets. The Power Supply operation is covered in ten sentences. These pages tell a little of the *what* and *where,* but little or nothing about the *why.* There is no discussion of the CPU or Display/Control on-board regulator schematics.





The Operator's Manual takes up 92 pages. The Introduction consists of three pages of Boolean logic and truth tables, two on electronic logic (gates), a paragraph on number systems, three short pages on the binary system, a page of generalities on computer programming, two pages on a simple program for adding two numbers, and a page on computer languages.

The part on Organization of the Altair 8800 includes a page with a block diagram but without any explanation whatever, five pages on the CPU, 2/3 of a page on memory, three sentences on the clock, and a third of a page on Input/Output. One sentence notes that "subtraction and division are implemented by inverse addition," yet inverse addition is left unexplained, by word or example; not even binary addition is shown. Figure 2-3, The Working Registers, is different enough from Figure 2-2, the CPU diagram, to cause confusion. Only a few of the CPU blocks are explained, leaving many others to the reader's imagination.

One Part in Detail

One part of the manual is explained in detail, telling how to load a simple program that adds two numbers, using LDA, MOV, ADD, STA, and JMP. The meaning of each mnemonic is explained, then the mnemonics are repeated along with the bit patterns they represent. This is then repeated, with octal equivalents of the bit patterns. The reader is then shown how to enter the program with the front-panel switches, all very clearly. A short section, 2/3 of a page, tells how to store the numbers to be added, in detail.

This whole section is written in the detail that the whole manual should be, with all the explanations the reader should be given. But perhaps this detail is given here because this is the most important part of the entire manual; without such detail, the builder can't figure out how to run a simple program, and thus won't be able to use the computer he's spent weeks or months abuilding.

The only other program in the entire manual is for binary multiply, unaccompanied by a single word of explanation.

The single page on Memory Addressing describes, in several sentences each, the various addressing modes for the 8800. The page of Operating Hints covers three items: proofread your programs; scatter NOPs through a complicated program so that new steps can be added if necessary; for debugging, use the SINGLE STEP switch and observe the status LEDs.

The rest of the manual, 45 pages, covers the instruction set. For each, the manual gives the mnemonic, bit pattern, what operation is caused by the instruction, what status bits are affected, and sometimes a brief example. The example assumes some knowledge of programming, as does the paragraph on operation. For DI, the operation paragraph reads, "Implementation of the DI instruction resets any interrupt flip-flop. This causes the computer to ignore any subsequent interrupt signals."

Again, this explains what and where, but not why. There are no programs, other than the two brief ones for add and multiply, to demonstrate the use of all these instructions, as there should be to help the reader write his own programs with them. Many readers won't be able to progress beyond using only a few of the basic instructions, unless they get some help, either in the form of sample programs, or a text on the use of this (or a similar) instruction set.

These 45 pages are fine for a programmer already familiar with assembly language, but the beginner will be almost totally lost.

An appendix of 5½ pages condenses the instruction set into tables.

Memory for the 8080

Since some memory is needed to make the computer work, let's take a look at the documentation for the 4K RAM board. The 2¼ pages on Theory of Operation explain the refresh operation, with schematic references, telling *what,* but not *why,* other than to say, after several paragraphs about which pulse goes where, that "the output of this circuit provides a 500 nanosecond, +12 volt pulse to the RAM IC's to accomplish the required access."

The errata sheet says diode D2 will be a 1N746A 3.3 volt zener, unless "your kit is supplied with the Intel C2107A's," in which case "this diode will be a 1N4733, 5v zener instead." Mine was a 1N4742A.

The last page contains a modification to the power supply, to add a paralleled pair of resistors, but nothing at all on why.

The "first impressions" must end here, because I'm waiting for MITS to return the 4K memory boards, and to send some software. Next time, we'll look into the programming, both in assembly language and in BASIC, and into more of the hardware, including interfaces.

Despite the Knocks

Despite all that can be said against the Altair, the 8800 is well on its way to becoming the best-known and most-used microcomputer kit of the decade. It may not be the best (the Sphere seems to have many advantages) nor the best supported (the Scelbi provides much more software help, in print), but it is today's bestseller. And unless some other company can fight this already well-established computer kit with a bigger and better advertising campaign (and not necessarily with a better computer kit), then the 8800, the 680 and future Altair computers will give MITS the micro-equivalent of IBM's continuing and overwhelming success.

Calculators in the Classroom

Pro: Calculators make tedious math fun, fast and accurate, educators and students agree. When used for creative problem solving, motivation in students seems spontaneous.

Con: Mechanization of fundamental classroom skills may leave kids unable to do simple math on paper. The cost for electricity or batteries may make operating the device daily too expensive.

by Deedee Pendleton

Conrad, a Washington, D.C., second grader, is 1 billion, 296 million seconds old right . . . now. Or, if you prefer, seven years, three months and five days. If you ask him how he knows, he'll tell you he figured it out on his calculator. If that sounds a little unsettling, relax. Conrad is getting a first-hand lesson in using his father's $40 electronic hand calculator at school. And although some parents are complaining the basics of education are being undermined by machines, the kids seem to love it.

Pocket math, as it's called, has been assaulted on all sides, but both the manufacturers of electronic hand calculators and progressive educators are anxious to see one in every classroom, if not one at every desk. Some first graders are already doing basic addition with calculators the minute their teachers feel they understand the principles, and high-school and college students are buying calculators as if they were radios.

Some calculators cost as little as $20, or about the same as some textbooks, and instructors say they could become required equipment in advanced math classes. The pocket-sized units are already replacing textbooks in elementary schools, and teachers are hoping that what once seemed to children a tedious labor may, through the calculator, become fun.

Opponents of calculators say that kids won't know how to count if their calculator batteries ever go dead, just as TV-oriented students no longer seem to know the basics of grammar and spelling. The device, critics contend, will make pencil-and-paper math obsolete.

But instructors who are using them take the opposite stand. They say that calculators stretch the student's interest, allow for more relevant kinds of problems (how far is it to the moon?) and increase motivation. Because of their speed and accuracy, calculators lend themselves to complicated problems previously avoided by grade-school teachers.

"One of the important uses of hand calculators is to enable children to solve more interesting problems, and to work out large divisions which would otherwise discourage them," says George Springer, an Indiana University mathematics instructor. Thus, oversimplified problem solving becomes unnecessary.

Teaching the basics before letting the child experiment with the calculator, many arithmetic teachers say, is essential for the machine's best use. "The hand-held calculator can be a very valuable tool, but only to an operator who understands the basic ideas, concepts and meanings behind the instantaneously generated answers it provides," says Frank S. Hawthorne of the New York State Education Department. Unlike the abacus, a calculator provides little or no help in learning computational skills.

Calculators will help children adjust more readily to a technological world, Springer says, and will make it easier for them to understand decimals, on which the metric system is based.

There is some opposition to the calculator in the classroom, admits Douglas Lapp, a Fairfax County (Va.) science curriculum specialist, but he says it isn't always valid. "Americans are particularly prone to think technology will offer easy solutions to everything, when in fact it simply solves existing problems, but does nothing automatically.

"The fundamental problem in math education still is that kids too often don't know the meaning of mathematical education and won't learn any more than they did by rote memorization," Lapp says. "We need to first give them concrete specific solutions as physical models for multiplication, which they can later transfer to concepts." Jill Horlick, an elementary-school math specialist, agrees. "The calculator doesn't think for you; it doesn't have a brain." She says that once her students can understand the theory of multiplication, they can adapt their knowledge to their imaginations. "Kids *normally* think about the universe; they love to manipulate large numbers because it makes them feel important. Why stop [the child] from thinking beyond those numbers just because he doesn't have the tools yet?"

New mathematical principles adapted for the calculator classroom are inevitable, Horlick maintains. More emphasis is placed on estimation, or on learning to judge which of the answers the calculator gives is reasonable. In addition, decimal placement becomes much more understandable, she says, because the calculator is able to provide answers of 6 to 12 digits, far beyond a young child's ability to calculate on paper. Children too often become bogged down in the complexity of a problem on paper, "and lose sight of the problem they are trying to solve," while the calculator eliminates the long rows of numbers usually associated with four-digit multiplication problems.

Douglas Grouws, a University of Missouri mathematics education instructor, holds that educators "must pay careful attention with regard to how we use [calculators] in the class-

. . . Calculators

room. I don't think availability will necessarily eliminate the need to be able to calculate by hand. However, the calculator may shift the emphasis away from proficiency in hand calculations to a greater emphasis on the meanings of the operations and when they can be appropriately applied." Grouws recommends using calculators in combination with basic arithmetic skills by, for example, providing them to students to check handworked problems.

But Horlick maintains that using the calculator for a combination of processes is essential. "You're defeating the purpose if you only used the calculator to check answers. The child wouldn't be learning to use the principles of the calculator."

The primary question, in her view, is, "Does the student know what he's doing?" Much more emphasis must be placed on, "What does it all mean?" than on "How fast can you get the right answer?" Those most opposed to calculators have gone so far as to ban them from the classroom, fearing that the device could become a crutch and keep students from learning the basic mathematical skills. Another argument for calculators, though, is that they make complex and realistic teaching exercises possible (how many cubic centimeters would it take to fill this room?). First graders, Horlick says, love to plan a family vacation, calculating costs of gas, motels and food.

A survey of teachers, mathematicians and laymen by MATHEMATICS TEACHER magazine has revealed that 72 percent of those polled opposed giving every seventh-grade student a calculator to use during his secondary education, but 96 percent agreed that "availability of calculators will permit treatment of more realistic application of mathematics, thus increasing student motivation."

In Virginia's Fairfax County, math teachers voluntarily agreed to permit high-school students to use calculators for homework and for some class assignments, but to forbid their use on tests unless every student in the class has one.

"With prices so low for calculators, it's no more a flight of the imagination to buy a calculator than it would be to buy a textbook," Springer says. When industry uses metrics and decimals exclusively, he adds, students taught to use calculators in school will be able to adapt quickly. There is one calculator for every nine Americans, and students who can't afford their own often borrow calculators from their parents. The device has become an essential part of training in statistics and computer science.

By 1976, the price of some calculators is expected to drop to as little as $10. If it does, the possibility of supplying public schools with them, and consequently incorporating them into elementary- and high-school math programs, may become very real. ☐

It appears as though Deedee Pendleton could have used a calculator herself when she wrote her article on "Calculators in the classroom." Her opening sentence states that Conrad, a Washington, D.C., second grader, is 7 years 3 months and 5 days old, and that this figures out to 1 billion 296 million seconds. If she had checked her second grader's results with her own calculator she would have seen that his age actually figured out to slightly less than 230 *million* seconds!

A billion seconds is actually quite a long time. One of the techniques used to impress upon the general public the differences between a million and a billion (particularly when the Federal budget is being discussed) is to point out that a *million* seconds is just over 11½ days, while a *billion* seconds is about 4 months short of 32 years!

William A. Robinson, P.E.
Solon Mills, Ill.

Reprinted with permission from Science News. Copywright 1975 by Science Service, Inc.

Tips for Buying A Pocket Calculator

By PETER WEAVER

"Miniature, electronic calculators could become as common on kids' desks as pens, pencils and rulers."

So says the superintendent of a major urban school system. As the price of these tiny mathematical wonders drops dramatically, more and more students will be able to afford them.

Right now, you can buy a basic, four-function model (add, subtract, multiply and divide) for as little as $19.95. The lowest price a year ago was closer to $90. Next year, the price for the least expensive models could drop below $15.

Whether a student should be using a calculator for doing homework depends on the math teacher you talk to. Some say the little machines are great and take a lot of the "dull, donkey work out of working with figures." Others say the calculators "could become a crutch and prevent the student from really learning fundamentals." Some compromise by saying "up to junior high school or so, students should do their homework the old way and only use the calculators to check their results."

In high school and college calculators can greatly speed up project work in math, science and business administration courses. But, in high school students probably need more sophisticated calculators, which can retain sums in a "memory" and can do percentages or square roots. These machines cost around $70.

Whether you're buying a calculator for a student going back to school or for yourself to use around the house (shopping, checkbook balancing), here are some things to look for:

●Keyboard. Are the keys hard to depress? Are they too close together for your fingers (pressing two down at the same time)? Do you know when you've registered a figure by a clock or a sense of touch?

●Number display. Are the numbers easy to read? Or are they in broken lines that tend to make eights look like zeros? Can you read numbers at an angle (some cheap models require reading head on)?

●Batteries. How long do the batteries last before changing? Some last only eight hours, other last 108 hours. Is an AC wall plug available?

●Logic Systems. Some calculators base their computing logic on a mathematical system, others base computing on a stacking system or algebraic system. Algebraic is easy for average users to understand because it works the way you would state a problem (100 minus 25 equals 75).

●Warranties. Most manufacturers give one-year warranties and are pretty good about fixing or replacing defective machines. But, who has to send the machine back to the manufacturer — you or the dealer? How long does it take for repairs? For the inexpensive, $19.95 calculators, it probably isn't worth it to have them repaired after the warranty runs out. It would cost too much. A few dealers are giving two-year warranties on machines costing $35 and up.

Don't buy extra mathematical functions you won't need no matter how exotic and prestigious they sound. It's a needless expense.

General Purpose Calculator Ratings

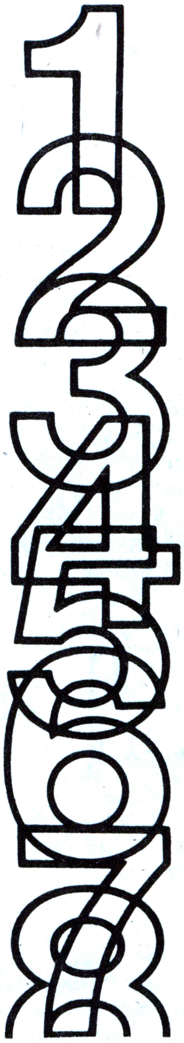
Consumer Reports rated general-purpose pocket calculators in their September 1975 issue. These are the units that are one step up in functions and features from the basic 4-function unit. One important message: DON'T BUY EXTRA FUNCTIONS UNLESS YOU NEED THEM. The article and reviews were excellent and we recommend looking at this issue before you invest in a pocket calculator.

Presented below are some of the performance factors that went into the ratings. The table is in order of overall rating.

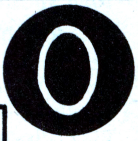
PERSONAL CALCULATORS: PERFORMANCE FACTORS

KEY: E, Excellent; VG, Very Good; G, Good; F Fair; P Poor

	KEYBOARD			DISPLAY		
GENERAL PURPOSE	Entry	Versatility	Keystroke count	Readability	Viewing angle	Brightness in office light
KEYSTONE 2030	E	G	G to VG	G to VG	E	VG
CORVUS 415	E	G	G	G	E	G
SEARS 5825	E	G to VG	E	G	G to VG	F
CASIO MINI-PRINTER	F to G	F to G	E	G	VG to E	G
SUMMIT SL8MR	VG	F	G	G	VG to E	VG
TEXAS INSTRUMENTS 2550	E	F	G	G to VG	G	G
MIIDA MC868M	F to G	G	E	G to VG	VG to E	G
APF MARK 30	F to G	G to VG	P to F	G to VG	VG to E	VG
SHARP EL8106	F	F to G	G	G to VG	VG to E	G
CASIO MINI-MEMORY	F to G	F	G to VG	F	E	G
COMMODORE GL987R	F to G	F	G	G to VG	VG to E	G
BOWMAR MX35	VG	F	P	G	VG to E	F
CANON LE81M	F to G	F to G	E	G	F	F 🄴
BROTHER 861	F to G	F	E	G to VG	F to G	VG
LLOYD'S EH8715	F to G	VG	P	G to VG	F to G	G
UNISONIC 739SQ	F to G	VG	P	G to VG	F to G	G
SINCLAIR CAMBRIDGE MEMORY	F	F	E	P	F	G
SCIENTIFIC						
TEXAS INSTRUMENTS SR50	E	VG	E	G	VG	G
HEWLETT PACKARD HP35	G	G to VG	VG to E	G	E	G
ROCKWELL 63R	F	VG	G	G to VG	VG	VG
COMMODORE SR1400	F to G	VG	G	G to VG	G	G
HEWLETT PACKARD HP21	VG	VG	G	G	F	F
SUMMIT SI90	E	G to VG	P to F	G	G	F
BOWMAR MX140	F to G	VG	P	G	E	F



Pocket Calculator Tricks!



Punch these problems into your pocket calculator, then turn it around (180°) to read the answer. For loads more of calculator problems, see the four calculator books in the *Creative Computing* Library advertisement.

An Ancient Arab Proverb:
0.1283 x 3 + 47 x 15

Where?
71 x 2 + 0.15469 x 5

And Then What?
121 x 57 + 0.25 x 16 ÷ 2

John Jackobs
Heidelberg College

The Stock Market Is Dropping!
$(508^2 - 16^2 + 5^2 + 2) \times 0.03$

Familiar Principle:
$(.844561)^{0.5}$

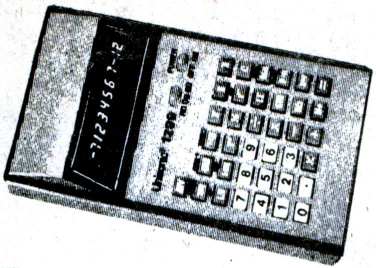
That's A Big One!
$50 \times 125^2 - 269^2 + 120$

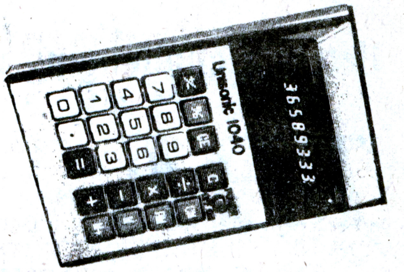
POOR HOUSE

If you buy 100,000 shares of IBM stock (ENTER 100000) on margin at $148.18 per share (ENTER x 148.18), pay $472 commission (ENTER + 472), and the price goes down 25% (ENTER x 0.25), what do you find yourself in?

David Ahl

7 POCKET CALCULATOR GAMES





From THE CALCULATING BOOK *by James T. Rogers, copyright © 1974 by James T. Rogers, reprinted by permission of the publisher, Random House, Inc.*

◆◆◆◆◆◆

To order *The Calculating Book,* see *Creative Computing Library* ad elsewhere in this issue.

◆◆◆◆◆◆ ← PAGE 78

Now your informative and versatile machine is going to tell you the month and day of your birth, together with your age. Hold on to your hat.

Take the number representing the month of your birth (January is 1, February, 2, and so on) and multiply it by 100. Add the date of your birth. Multiply by 2, add 9, multiply by 5, add 8, multiply by 10, subtract 422, add your age and subtract 108. The result will be a five-digit number; the first digit will tell the month of your birth, the next two the date of your birth (as, say, 08 if it was before the 10th of the month) and the last two your age.

◆◆◆◆◆◆

Your calculator can give you advice on driving. Suppose you are cruising along at 85 miles per hour and you want to know what advice your calculator would offer in that situation. Divide 85 by 79.069767 and read the answer upside down. (If you are driving that fast, maybe it would be better if you put yourself upside down and left the calculator right side up.)

If you want to talk back to the calculator, giving it either an instruction or an opinion of its character (depending on your mood), enter 7334 and read the answer upside down.

◆◆◆◆◆◆

Reach in your pocket and pull out whatever change is there. Count the amount, and with the total as the starting point (used as a whole number, omitting the decimal) do the following things: multiply by 10, add 1, multiply by 2, add 21 and multiply by 5.

The result will be a number ending in 15. Discard the 15 and subtract 1 from what is left. The answer will be the amount of change you started with.

Imagine a set of chips, each of which bears a number from 0 through 9. They are lined up in the following order: 6328907154.

Problem: Without changing the order except by moving digits from end to end (making, for example 4632890715), find the two groups that can be multiplied to produce the third group. Here again you can put your calculator through a multiplication drill. (Hint: One multiplier has three digits and one has two; the answer has five.)

Answer: 715×46=32890.

◆◆◆◆◆◆

Take a number having any reasonable number of digits (as many as six, say) and go through the following abracadabra:

Multiply by 2.
Add 4.
Multiply by 5.
Add 12.
Multiply by 10.
Subtract 320.

The result will be a number ending in one or more zeroes. Drop them and you will be left with the number that you started with.

Suppose you start with 52871. (You'd better write it down.) Then: 52871 × 2 + 4 × 5 + 12 × 10 − 320 = 5287100.

Starting with any three single digits, as long as they are different from one another, see if you can find a combination besides 6 × 21 where the product contains the digits that you multiplied (126 in this case).

It might strike you that this is a search for a needle in a haystack. In fact, however, there is only one combination to be found. (Hint: One of the multipliers is 51. You therefore can limber up your calculator on some trial multiplications of 51 and the digits from 1 through 9 until you find a combination that gives you an answer containing 5, 1 and the number you multiplied 51 by.)

Answer: 3 × 51 = 153.

You can drive yourself a bit closer to distraction by trying the same thing with four digits. In this case, as with three, you can arrange them in any way (4716, for example, could be 4 × 716, 47 × 16, 471 × 6, 7 × 416 and so on).

The possible solutions are: 8 × 473 = 3784; 9 × 351 = 3159; 15 × 93 = 1395; 21 × 87 = 1827; 27 × 81 = 2187; and 35 × 41 = 1435.

◆◆◆◆◆◆

The number 3025 displays a remarkable quirk when it is split into two parts, 30 and 25. Add the two parts and square the result. ■

Here's a challenging new 2-person game to play on your pocket calculator.

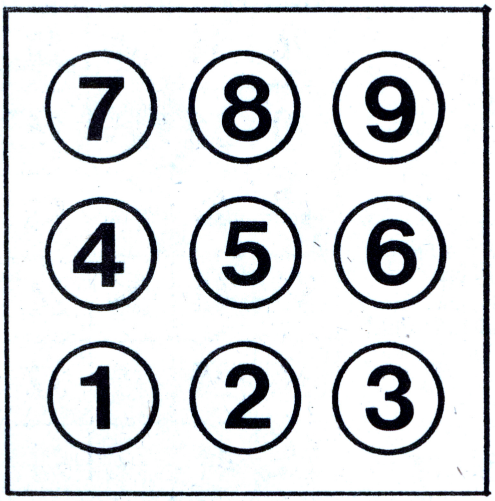
THE KEYBOARD GAME

by L. D. Yarbrough

Suppose someone approaches you, electronic calculator in hand, and suggests that you play the following game:

"We start by selecting a number to enter into the calculator—say 100. Suppose I choose the number: then you pick a key, from 1 to 9. You subtract that number from the total. Then I pick a key, only my key must be next to your key on the Keyboard. I subtract that, then you pick a key next to mine, and so on. The first one to turn on the 'minus' sign loses the game."

For example, if the *4* key is chosen, the next player may subtract only 1,2,5,7, or 8 from the total. If he chooses *5*, the other player may choose any key *except 5*. If he chooses *1*, the next key chosen must be *4, 2,* or *5*. Choosing a key always means subtracting the value of that key from the total.



Before reading further, I suggest you have some fun playing the game with a friend. If you don't have a calculator, try pencil and paper. The calculator just makes it go faster. And if you play with a calculator, remember always to hit the (—) key after *every* number key.

The winning strategy for this game is unusual in a way which I will explain shortly. To appreciate just how unusual it is, though, let's consider the solution of a related Keyboard game which is much simpler to analyze. Suppose we take away any restrictions on which of the keys (1-9) can be used at each turn. Now the winning method will soon become easy to see: just subtract the right-hand digit from the total. That, of course, produces a zero in that position, unless there was already a zero there. So if, at your turn, the total is a multiple of ten—especially if it is exactly zero!—you lose. Otherwise, you win. Now this means that the player who chooses the original total can win simply by choosing a multiple of ten, and subtracting the right-hand digit whenever it is his turn to play.

This situation holds in a very large number of games in which two players play alternately: the selection of the initial conditions of the game is quite often enough to win the game. NIM is a classical example. The literature of recreational mathematics is filled with other examples.

Now back to our original Keyboard Game. In his game, as we shall see, the player who makes the choice of the first *key* has a tremendous advantage. If we assume that the initial entry into the calculator is large enough—say greater than 15—the player who chooses the first key can announce his choice to the whole world, let the other player choose whatever starting total he wishes, and still win the game. The winning strategy depends on the position of the first key, not on its numeric value!

We can develop a winning strategy for any given initial total by building up a table, starting at zero, and calculating the winning keys as a function of currently available keys and winning strategies for lower totals. Figure 1 shows the winning keys for totals up to 100. In this figure, "T" appearing at a given row and column means, "For the current total equal to this row number, the key at the head of this column is a winner, if we can get to it." "F" designates a key choice which gives our opponent a chance to win. For example, if the total is 10, *2* is a winning key: it reduces the total to 8, and the only winning keys for 8 are the *7* and *8* keys, which are on the opposite side of the keyboard. *3* is a loser; it reduces the total to 7 and leaves the *6* key to our opponent.

From Fig. 1 it is clear that the corners of the keyboard, especially the *1* and *3* **keys**, offer winning opportunities for all initial totals above a certain limit. Choosing any other key nearly always loses because our opponent can grab a corner key and keep returning to it no matter what we try. For small totals we need to be careful because the winning keys are not always in the corner anymore.

What if we include the *0* key? If we give it its nominal value, the game becomes very simple: choose that key and keep returning to it; our opponent is the only one who subtracts anything from the total. So give it a value of 10. You will enjoy investigating this variant of the Keyboard Game for yourself. Actually, it has several variants: on some calculators the *0* key is under the *2* key, on others it is under the *1*, etc. In this last version, with *0* under *1*, there is *one* key which is a winner for all totals above 35. I leave it to you to figure out which one it is.

	1	2	3	4	5	6	7	8	9
0	F	F	F	F	F	F	F	F	F
1	T	F	F	F	F	F	F	F	F
2	T	T	F	F	F	F	F	F	F
3	F	F	T	F	F	F	F	F	F
4	T	F	T	T	F	F	F	F	F
5	F	F	F	F	T	F	F	F	F
6	F	F	T	F	F	T	F	F	F
7	T	F	T	T	F	T	T	F	F
8	F	F	F	F	F	F	T	T	F
9	T	F	F	F	F	F	T	T	T
10	T	T	F	T	T	F	T	T	T
11	F	F	T	F	F	F	F	T	T
12	T	F	T	F	F	F	F	F	T
13	T	F	F	F	F	F	T	F	T
14	T	F	T	F	F	F	F	F	F
15	T	F	T	F	F	F	F	F	F
16	T	F	T	F	F	F	F	F	F
17	T	F	T	F	F	F	F	F	F
18	T	F	T	F	F	F	F	F	F
19	T	F	T	F	F	F	T	F	F
20	T	F	T	F	F	F	T	F	F
21	T	F	T	F	F	F	T	F	T
22	T	F	T	F	F	F	T	T	T
23	T	F	T	F	F	F	T	T	T
24	T	F	T	F	F	F	T	T	T
25	T	F	T	F	F	F	T	T	T
26	T	F	T	F	F	F	T	T	T
27	T	F	T	F	F	F	T	F	T
28	T	F	T	F	F	F	T	F	T
29	T	F	T	F	F	F	F	F	T
30	T	F	T	F	F	F	F	F	T
31	T	F	T	F	F	F	F	F	F
32	T	F	T	F	F	F	F	F	F
33	T	F	T	F	F	F	F	F	F
34	T	F	T	F	F	F	T	F	F
35	T	F	T	F	F	F	T	F	F
36	T	F	T	F	F	F	T	F	T
37	T	F	T	F	F	F	T	F	T
38	T	F	T	F	F	F	T	F	T
39	T	F	T	F	F	F	T	T	T
40	T	F	T	F	F	F	T	T	T
41	T	F	T	F	F	F	T	T	T
42	T	F	T	F	F	F	T	F	T
43	T	F	T	F	F	F	T	F	T
44	T	F	T	F	F	F	T	F	T
45	T	F	T	F	F	F	T	F	T
46	T	F	T	F	F	F	F	F	T
47	T	F	T	F	F	F	F	F	T
48	T	F	T	F	F	F	F	F	F
49	T	F	T	F	F	F	T	F	F
50	T	F	T	F	F	F	T	F	F
51	T	F	T	F	F	F	T	F	T
52	T	F	T	F	F	F	T	F	T
53	T	F	T	F	F	F	T	F	T
54	T	F	T	F	F	F	T	F	T
55	T	F	T	F	F	F	T	F	T
56	T	F	T	F	F	F	T	T	T
57	T	F	T	F	F	F	T	F	T
58	T	F	T	F	F	F	T	F	T
59	T	F	T	F	F	F	T	F	T
60	T	F	T	F	F	F	T	F	T
61	T	F	T	F	F	F	T	F	T
62	T	F	T	F	F	F	T	F	T
63	T	F	T	F	F	F	F	F	T
64	T	F	T	F	F	F	T	F	T
65	T	F	T	F	F	F	T	F	F
66	T	F	T	F	F	F	T	F	T
67	T	F	T	F	F	F	T	F	T
68	T	F	T	F	F	F	T	F	T
69	T	F	T	F	F	F	T	F	T
70	T	F	T	F	F	F	T	F	T
71	T	F	T	F	F	F	T	F	T
72	T	F	T	F	F	F	T	F	T
73	T	F	T	F	F	F	T	F	T
74	T	F	T	F	F	F	T	F	T
75	T	F	T	F	F	F	T	F	T
76	T	F	T	F	F	F	T	F	T
77	T	F	T	F	F	F	T	F	T
78	T	F	T	F	F	F	T	F	T
79	T	F	T	F	F	F	T	F	T
80	T	F	T	F	F	F	T	F	T
81	T	F	T	F	F	F	T	F	T
82	T	F	T	F	F	F	T	F	T
83	T	F	T	F	F	F	T	F	T
84	T	F	T	F	F	F	T	F	T
85	T	F	T	F	F	F	T	F	T
86	T	F	T	F	F	F	T	F	T
87	T	F	T	F	F	F	T	F	T
88	T	F	T	F	F	F	T	F	T
89	T	F	T	F	F	F	T	F	T
90	T	F	T	F	F	F	T	F	T
91	T	F	T	F	F	F	T	F	T
92	T	F	T	F	F	F	T	F	T
93	T	F	T	F	F	F	T	F	T
94	T	F	T	F	F	F	T	F	T
95	T	F	T	F	F	F	T	F	T
96	T	F	T	F	F	F	T	F	T
97	T	F	T	F	F	F	T	F	T
98	T	F	T	F	F	F	T	F	T
99	T	F	T	F	F	F	T	F	T
100	T	F	T	F	F	F	T	F	T

Figure 1

A POWERFUL PROBLEM

Some years ago, a mathematician employed by one of America's leading aircraft manufacturing companies got into the habit of 'number-doodling'. Number-doodling is where you start from some particular number (a telephone number, a birthdate, a car number, etc.) and see what sort of mathematical peculiarities can be developed from the number.

It turned out that the mathematician lived at 153 Westpark Street. After some preliminary doodling with this number, he noticed that if he added together the third powers of the constituent digits, he would arrive back at the number 153. That is:

$$1^3 + 5^3 + 3^3 = 153$$

When he tackled his four digit telephone number in the same way, he was agreeably surprised to find the same effect. His telephone number was 8208. Summing the fourth powers of the constituent digits gives the total 8208. That is:

$$8^4 + 2^4 + 0^4 + 8^4 = 8208$$

Armed with these two fascinating pieces of number-doodling, he began to treat methodically every number connected with his existence in the same way. That is, for a number with *n* digits, each of the constituent digits would be raised to the *n*th power and then summed, in the hope that the sum was equal to the original number.

Surprisingly, the zip code of the area in which he lived, 54748, was notable for exactly the same property. That is:

$$5^5+4^5+7^5+4^5+8^5=54748$$

And his car's registration number, 548834, displayed exactly the same peculiar property:

$$5^6+4^6+8^6+8^6+3^6+4^6=548834$$

And his employee number at the aircraft company, too, was equally odd. Who else but a mathematician could see any beauty in a number like 1741725? Still, you must admit that it is rather interesting that:

$$1^7+7^7+4^7+1^7+7^7+2^7+5^7=1741725$$

Three other numbers that formed part of his life also had this property. His bank account number (24678050), his driving licence number (146511208), and his Social Security number (4679307774). Thus:

$$2^8+4^8+6^8+7^8+8^8+0^8+5^8+0^8=24678050$$

$$1^9+4^9+6^9+5^9+1^9+1^9+2^9+0^9+8^9=146511208$$

$$4^{10}+6^{10}+7^{10}+9^{10}+3^{10}+0^{10}+7^{10}+7^{10}+7^{10}+4^{10}$$
$$=4679307774$$

Would the reader care to search for other numbers, up to ten digits long, that have the same properties? All told, there are no more than 15 of these additional numbers, assuming that you exclude all the one-digit numbers. One last question: what was the name of the town in which our number-doodling mathematician lived?

Is your number up?

Everyone's attention wanders once in a while, but the lapse could be fatal at the top of the stairs, on the road or while using a power tool.

J. R. Block of Hofstra University has researched a new technique to single out people who have trouble paying attention. The method can be used to predict a person's performance on tasks requiring uninterrupted attention.

Block's method involves an illuminated board with randomly arranged numbers that glow in five different colors. The task is to locate the numbers in order. Most people will find each in 10 to 20 seconds. He says slow performance indicates momentary attention gaps, but very rapid performance may indicate a person who sacrifices accuracy for speed. Both extremes can be dangerous.

34	19	42	54	45
26	16	39	28	57
40	35	14	56	30
12	29	44	51	23
50	43	36	24	11
37	20	55	32	47
25	41	17	53	38
13	22	48	10	58
52	18	21	31	46
27	49	33	15	59

* * *

Reprinted with permission from Games & Puzzles. Copyright 1975 by Edu-Games (U.K.) Ltd.

Beating the Game

Game theory compares blackjack systems and proposes to teach a computer backgammon

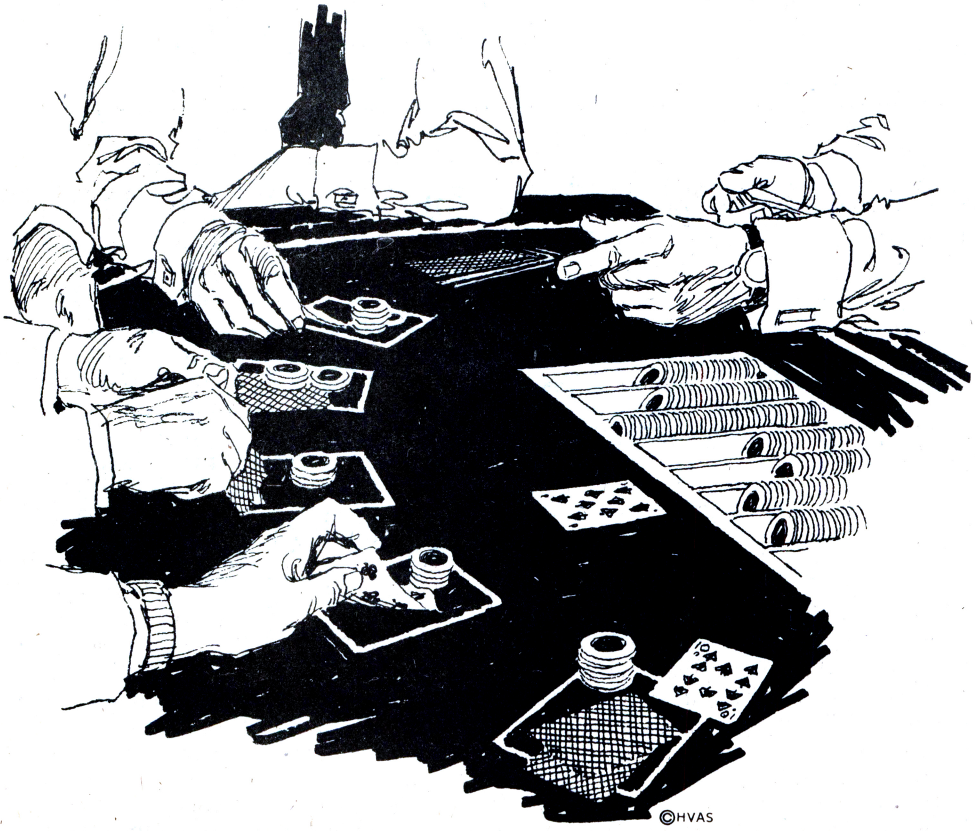
by Dietrick E. Thomsen

The man who broke the bank at Monto Carlo is a musical fantasy that grew out of the avid interest many people have in the things that happen on green baize tables. The man who, according to folklore, was told not to return to Las Vegas because he had won too much money there is real. He is Edward Thorp, a professor of mathematics at the University of California at Irvine.

Lately Thorp has been looking for the best way, in a theoretical and practical sense, to beat the blackjack table. He has devised a way of comparing the several blackjack systems against each other and a theoretically best possible system. At the same time his interest has turned to that ancient, but recently trendy, game, backgammon. He shared some of his latest insights on these topics with fellow mathematicians at the recent National Mathematics Meeting at Washington.

The blackjack systems depend on counting the cards as they fall. As play proceeds, the deck is depleted, and with the fall of each card the player's expectation of success changes. Removal of different denominations from the deck changes the expectations in different amounts. Removal of two nines does not have the same effect as removal of three fives.

From the way the expectations change, a particular numerical value can be assigned to each denomination of card. As the cards fall from the deck a running total of these numbers is kept. Different systems assign different numbers to different denominations. They also differ in how they use the running value total and the number of cards remaining in the deck. There are various combinations of addition and division. There is also a difference in whether the system keeps a separate count of aces. Aces have two values in blackjack, 1 and 11, and therefore some system makers like to tally them separately. The result of all this arithmetic is used to advise the player how to bet.


©HVAS

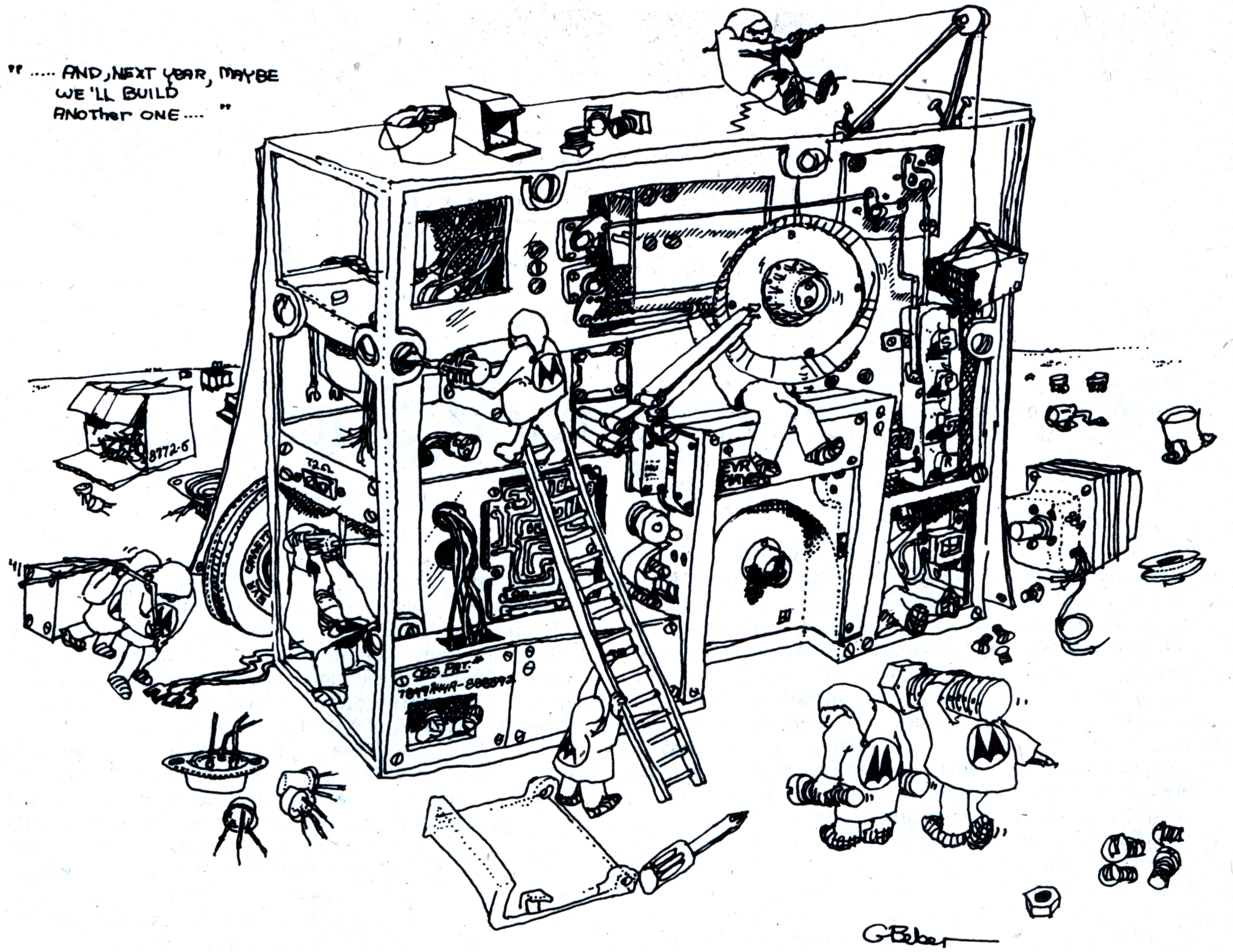
The question Thorp set himself was whether there is some method of comparing the different strategies without doing a massive computer simulation of a million hands. He finds one and he finds a criterion to compare them with each other and see how close they come to a theoretically possible optimum system.

First he needs a definition of "more advantageous." It may seem obvious that it means a greater chance of winning, but the case is complicated because a given system may give a greater expectation of winning when the play is in a particular stage, but it may be surpassed by another in a different situation. The final working definition of advantage is a system that gives at least as good a chance of winning over as wide a range of situations as an alternative with at least no more risk to the player.

Thorp finds that he can compare the quality of systems by defining an expectation function for each one that expresses its relative betterness. The expectation changes as play proceeds. It depends on the fraction of cards remaining in the deck, and it varies as they fall.

Graphically the expectations define a surface called a simplex, and the falling of cards causes motion from point to point on this surface as the expectation changes. Working with the geometry of the simplex Thorp can compare system to system, and he finds that he can define a single number, which he designates with the symbol lambda, that expresses a system's betterness relative to others and its closeness to a theoretically possible optimum system. Thus he has an analytic method for ranking blackjack systems and no longer has to simulate a million hands on a computer to compare them. But he does not tell us which is the best possible system.

Backgammon is among the most ancient games. A set dating to 2600 B.C. has been found. From the game theore-


" ..... AND, NEXT YEAR, MAYBE WE'LL BUILD ANOTHER ONE .... "
8772-6
T.R.O.
CBS PRI.
GBeber

tician's point of view, Thorp says, the hope of solving the problems that it presents is quite good.

The game consists of a ladder of 26 cells. Of these, 24 appear on an actual backgammon board. For the analysis, Thorp adds, the two "off the board" spaces at each end into which counters that have successfully completed their journeys are put. Each player's counters start at one end of the board, and the object is to get them all across the board and off it, passing the other player's counters coming in the opposite direction, before he gets his across. Moves are determined by throwing dice.

There is an important complication. If counters of both players arrive in the same cell, there are situations where one can be sent back to the beginning of its trip. This possibility of repeated restarts makes the game in a theoretical sense potentially infinite. In principle a backgammon match could last forever. It is "a fact which will impede analysis slightly," Thorp concedes.

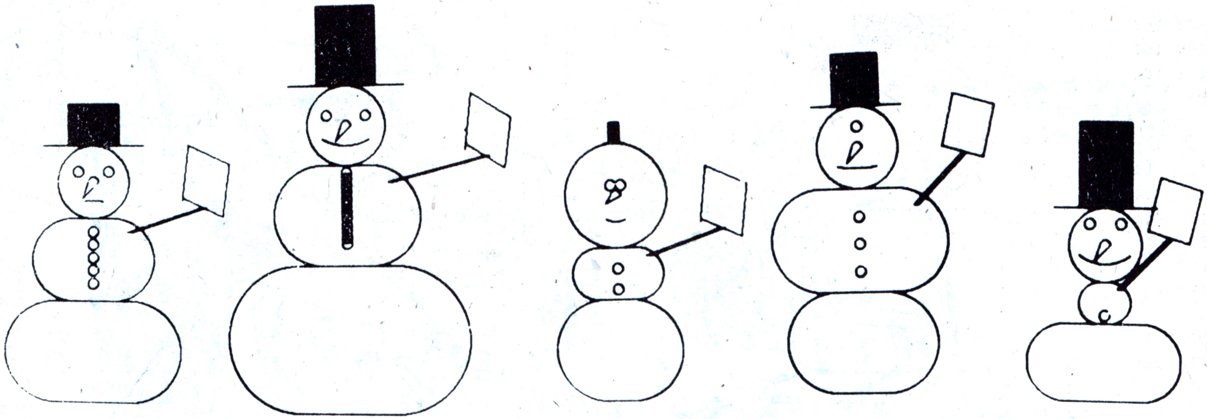
The way to analyze the game is to set up partial models that are simplified, removing complexities of the real game, especially the one that makes it infinite, and then gradually to add back the complications. In Model I each player has one counter, and the bounce-back rule is suspended so that the counters can freely pass each other. When this is properly set up it produces a game of 167 steps. The first to reach step 167 wins. Computer simulation shows that when goals are equal, the first player to roll has a slight advantage, but this declines toward even chance as play proceeds. This is a very crude approach to a real backgammon game, but it leads to interesting insights, Thorp says.

In Model II one sets up an end game. Again there are two counters, but they have already passed each other so there is no further chance that they could be sent back to their starting points. This too is a finite game and is amenable to solution.

One complication of the real game is the doubling cube. As the game proceeds, if one player gains a certain advantage, he can use the doubling cube to double the stakes. This changes the consequences for the loser and alters the expectations and strategy of play. Recursion schemes can be devised to solve both Model I with the doubling cube and the end game with the doubling cube. (A recursion scheme is a system for calculating a series of related values. Knowing the first number in the series and the recursion scheme you can calculate the second. Putting the second number into the recursion scheme gets you the third. And so on.)

In actual backgammon play it is possible that the game might come down eventually to Model I or Model II, but these highly restricted situations are still far from the complexity of a full game, in which each player has several counters on the board at once and the bounce-back rule can operate. Still enough has been learned so far for Thorp to conclude that backgammon "can be played better by computer than by any person." But suppose the computer refuses to go to Reykjavik? □

Reprinted with permission form Science News. Copywright 1975 by Science Service, Inc.



Computer-Planned Snowmen

Robert S. McLean
Ontario Institute for Studies in Education
Toronto, Canada

Computer models are useful for testing theories. If you have a model of a process or structure expressed as a computer program, you have a powerful tool. A computer can then produce many different instances of the modeled process through the use of a range of parameter values. It will produce the results of the model without being influenced by extraneous notions of what the outcome should be. In addition to investigating the range of applicable parameter values, one can even push those values beyond "reasonable" limits and observe the results. The computer tirelessly shows the results of the chosen conditions without requiring very many input resources. If the result of the model is a picture, so much the better; its adequacy can often be judged visually by the user very rapidly. What is more, any pictorial outputs can be very entertaining when they illustrate weaknesses of the model or data, thereby producing very unusual pictures.

Computer modeling at first seems like such a high-powered idea that it would be hard to apply for fun. Maybe it ought to be reserved for sending men to the moon, finding petroleum resources, or managing large construction projects. Although these projects use modeling, simple things that an individual does can also benefit from these techniques. Consider building a snowman, for instance. Here is an important problem for the individual which can be solved with the power of modern computing.

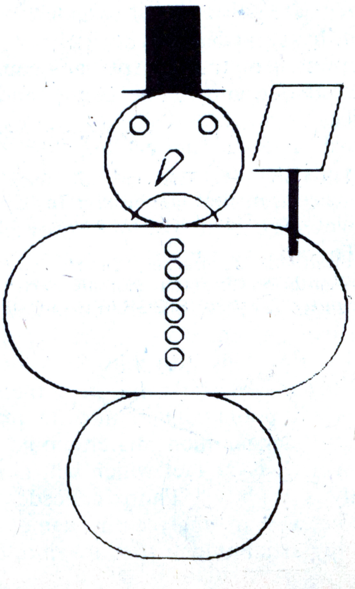
By choosing the proper aspects of a snowman as parameters for our model, we can use the computer to draw pictures of the resulting snowmen. We can observe the shape of hundreds of combinations of parameter values and select the most pleasing one for implementation with real materials. In addition to increasing the range of choice available, this procedure has several other advantages. Most of the design process as well as some of the construction effort is no longer at the mercy of the weather. Much of the work can be done in locations not previously suitable for this activity (home, office, school room, etc. were not suitable places for snowman construction activity previously). Valuable resources are not squandered in making real prototypes since one will have a better idea of the outcome of the process before starting to use these materials. Thus, computer-planned snowmen become feasible in areas where snow is in short supply. One can appreciate that the benefits are many.

What are the parameters of use in modeling a snowman? The reader may wish to propose his own set; for purposes of illustration, we provide a suggested set that were used to design the accompanying illustration. Our standard snowman will consist of three balls of snow, referred to as B1 (on bottom), B2 (middle), and head. In addition to the features of these components, we will add a hat and a broom. This is the economy version, since it could be given many more accessories, such as scarves, cigars, glasses, etc. These are left as an exercise for the reader.

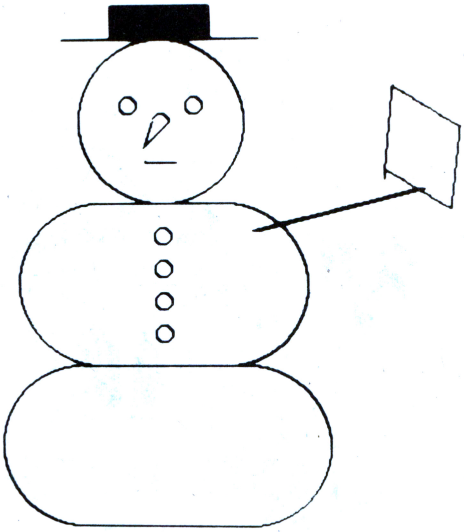
Snowballs 1 and 2 are not necessarily round; they often become somewhat flattened by the load above. Thus, two parameters specify these balls: the radius of the two half circles that form the curved part and the length of the flat parts on top and bottom of the ball, RAD and SIZ respectively, giving the first four parameters, B1SIZ, B1RAD, B2SIZ, and B2RAD. The middle snowball has, in addition, some number of buttons down its front (NBUT). These are spaced out over the vertical extent of B2.

In the world of economy snowmen, heads are always round; hence, there is one head size parameter, HEAD, the radius of the head. Five other parameters are used to place the eyes, nose and mouth on the head. In this version of the model, the nose is always placed in the center of the head and always has the peculiar (carrot-like) shape shown. The sole parameter available here is the length of the nose, NOSE. The eyes are located symmetrically above the center

```
B1SIZ   50
B1RAD  100
B2SIZ  200
B2RAD  100
N BUT    6
HEAD    80
NOSE    30
SMILE   50
SMIL2   55
EYEHT   40
EYEWD   40
HATWD   60
HATHT   50
BROOM  100
BDISP    0
```



```
B1SIZ 200
B1RAD 100
B2SIZ 150
B2RAD 100
N BUT   4
HEAD  100
NOSE   30
SMILE  20
SMIL2 200
EYEHT  20
EYEWD  40
HATWD 120
HATHT  20
BROOM  50
BDISP 200
```



of the head along a horizontal line EYEHT units above the center of the head. They are separated by EYEWD units, the interoccular distance.

The mouth is specified by two parameters, where SMILE gives the length of an arc used to denote mouth, and SMIL2 gives the radius of curvature of that arc. Since models should strive for generality in their parameters, SMIL2 may be negative as well as positive. If positive, the center of the arc is above the mouth, and if negative, it is below the mouth. Thus, we obtain the relationship that a frown is a negative smile! The mouth is always placed halfway between the nose and the chin, again in aid of simplicity.

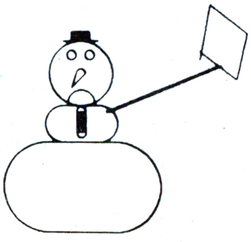
The remaining four parameters specify the two accessories, the hat and the broom. HATWD and HATHT specify the width and height of the hat; the brim is always twice as wide as the hat. The broom is perhaps the most difficult part of the model and as can be seen looks more like a shovel (a kind interpretation) or a strike sign (less kind). It surely leaves room for the reader to improve the model. The parameters used here are BROOM, giving the length of the handle, and BDISP, giving the distance that the broom handle upper end is displaced to the right from the lower end. Some very crude modeling results in the funny quadrilateral that looks more like a hoe in some cases.

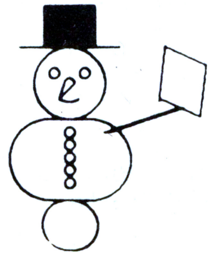
The reader can, no doubt, suggest refinements and is encouraged to do so. The inspiration for this program came from Chernoff's faces (1971) and an adaptation of them that is used to adjust the parameters of oral surgery and to illustrate the various ways in which facial features might be rebuilt (Eisenfeld, et al, 1974). But that's pretty serious business. For the amateur simulator, there are many other possibilities.

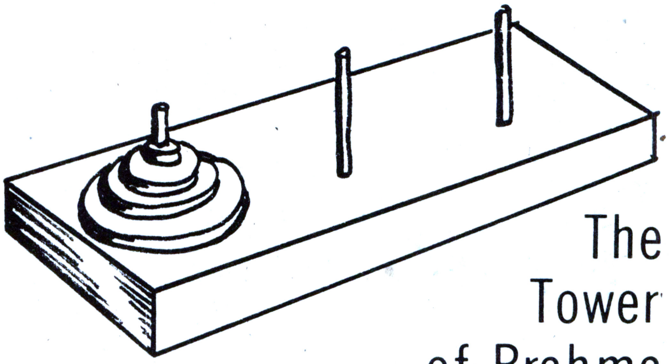
REFERENCES

Chernoff, H. The use of faces to represent points in n-dimensional space graphically. Tech. Report No. 71. Department of Statistics, Standford University, Stanford, California, December 1971.

Eisenfeld, J., Barker, D. R., and Mishelevich, D. J. Iconic representation of the human face with computer graphics. *Computer Graphics* (SIGGRAPH—ACM) 1974, 8(3), 9—15.







The Tower of Brahma

In the great temple at Benares beneath the dome which marks the center of the world, rests a brass plate in which are fixed three diamond needles, each a cubit high and as thick as the body of a bee. On one of these needles, at the creation, God placed sixty-four discs of pure gold, the largest disc resting on the brass plate and the others getting smaller and smaller up to the top one. This is the Tower of Brahma. Day and night unceasingly, the priests transfer the discs from one needle to another, according to the fixed and immutable laws of Brahma, which require that the priest on duty must not move more than one disc at a time and that he must place this disc on a needle so that there is no smaller disc below it. When the sixty-four discs shall have been thus transferred from the needle on which, at the creation, God placed them, to one of the other needles, tower, temple, and Brahmans alike will crumble into dust, and with a thunderclap, the world will vanish.

If the priests were to effect one transfer every second, and work twenty-four hours a day for each day of the year, it would take them 58,454,204,609 decades plus slightly more than 6 years to perform the feat, assuming they never made a mistake—for one small slip would undo all their work.

How many transfers are required to fulfill the prophecy?

A. Set up a program which allows the user to move disks by hand. You can try your ingenuity at drawing the result by some sort of plot or graph.

B. At least verbally, indicate how one would proceed in any arbitrary case (5 disks, 6 disks, etc. 64 is too much to try!).

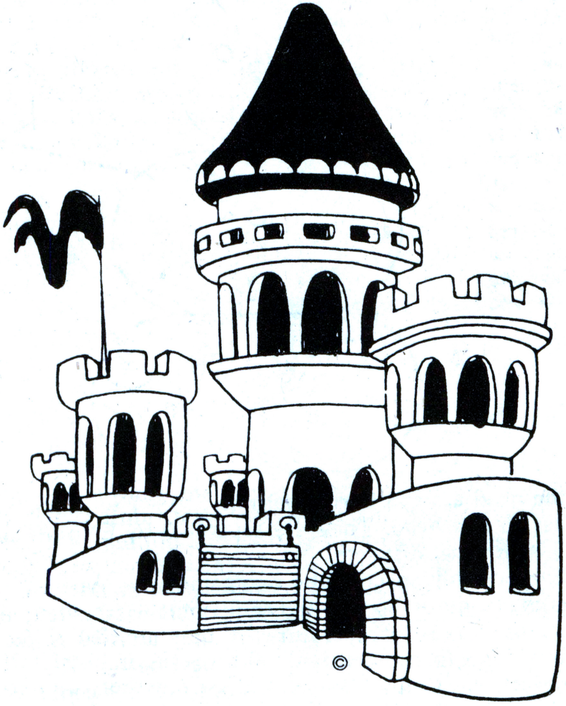
C. The monks, like monks everywhere, never eat, sleep, rest or die. If they have been moving one disk per second since the world began, how long will the total age of the universe be on Thunderclap Day?

Note: Prove that a game of N disks can be played in 2^N-1 (2-to-the-N, minus 1) moves.

D. Analyze the problem in this way: to move 5 disks, what kind of 4-disk moves are required? How do the "from" and "to" of these moves relate to the "from" and "to" of the 5-disk level?

DISCS	MOVES	DISCS	MOVES
1	1	6	63
2	3	7	127
3	7	8	255
4	15	9	511
5	31	10	1023

The Sleeping Queued T



by Jack Ludwig and Jack Le Baron

In tasks long PASCII'd there ran a REG/AL Unit and QUEUE$_n$*, who Swapped to each other every cycle of their *LIBs,* "Would that we had *HYBRID DATA!"* and yet they had none. But it happened once that when the QUEUE$_n$ was algorithming, there came an Freg* out of the DIOCS,* and he spooled on the cassette and said, "Thy list shall be fulfilled. Before a shift has gone by, thou shalt bring data into the Core L'd.*"

And as the Freg foretold, so it happened; and the QUEUE$_n$ buffer'd ASCII Data so beautiful that the REG/AL unit could not contain IMSelf* for JOYVIAL,* and he ordained a great Access. Not only did he bid to it his Chains, Pointers, and Indices, but also the I/Os Drivers, that they might be kind and Parity prioritze to the Data. There were seventeen of them in his Low partition, but as he had only provided hexidecmal based golden page sets for them from which to interleave, one of them had to be left out. However, the Access was initiated with all linear positioning; and as it drew to an end, the I/Os Drivers stacked forward to present to the ASCII Data their wonderful gifts: one bestowed virtual, one priority, a third registers, a fourth gave binary and so on, whatever there is in the Core L'd to list for.

And when fifteen 01 to OF of them had their say, in came a little five level baud OT, the uninvited seventeenth (11)$_{HEX}$, churning and overflowing to reallocate herself, and without indirect or relocatable loading, thrashed out of the background, "In the fifteenth$_{10}$ *SHARE* of her page your ASCII Data shall EBCDIC herself communicating with Infernal Bubble Memories and fall down abended.

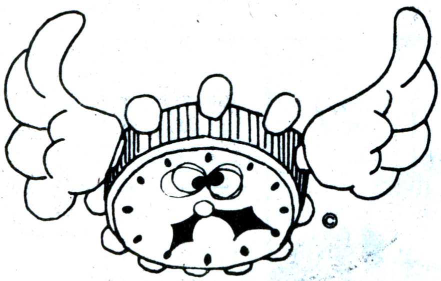
And without *zeroing* one more word she turned away, reset the acumulator and left. Everyone re-verified her saying, when the real first of the sixteen$_{16}$ I/Os Drivers came forward (00), for she had not yet bestowed her gift, and though she could not do away with the ill timed prophecy, yet she could Fail Soften it. So she said, "This Data shall not abort, but fall into a Wait State for MEGA time slices and GIGA cycles."

Now the REG/AL Unit, being designorous of saving his baby Data even from this misfortran, gave commandment that all the TP devices in his network should be micro-coded for ASCII only. The young Data was enpolled and went to elementree school, became a data element, joined other elements and was an activist in a device file table and her participation earned her a poll list record and on and on. Thus the maiden Data grew up, adorned with all the gifts of the I/Os Drivers; and she was so lively, optimal, swift, and kind and clever that no task who used her could help linking her. She matured and grew buffers, and a parallel direct memory bus. She was a real accumulator, and could she bump storage. It happened one batch, she being already multishifts old that the REG/AL unit and QUEUE$_n$ displayed at NCC, and the sacred Data was left behind in the CACHEL. She wandered about into all the modes, subindices, and VTOCS, and into all the directories and libraries, as the fancy took her, till at last she came to an old translator. She climbed the narrow winding search argument which led to a little displacement with a rusty index key sticking out of the base lock. She turned the key, and the INSTR'OP'D, and there in the CRYPTIC MODULE ROM sat an old I/Os Driver with bent folded, mutilated and worst of all spindled EBCDIC, where she was diligently modulating her hardwired messages.

"Good Day, Driver," said the Maiden Data. "What are you doing?" "I am Modulating." answered the old (11) Driver, nodding her head.

"What is that thing that twists round so briskly?" asked the Maiden Data, and taking the Proms onto her associative lists, she began to modulate, but no sooner had she touched it than the ill timed prophecy was fulfilled, and she EBCDIC'd her linker with it. In that very Pica second she fell back upon the stack of file addresses (SOFA) that stood there, and lay in a deep irrecoverable Wait State. She has become a *QUEUED T*ask Control Block gone to sleep, and this wait state fell upon the whole CACHE L Memory; the

REG/AL unit and QUEUE$_n$ who had returned and were in the great BALR, fell fast ASWAP, and with them the whole foreground. The MACROs in their calls, the General Regs in the Core, the P regions on the drum, the files in the vol, the very lights that flickered on the console became still, and waited like the rest; and the pack on the spindle ceased rotating, and the controller who was going to RE-IPL the floppy micro disk for some mistake made, let it go and joined the wait. And the Real Time Clock ceased, and not a message fell from the TTY's about the computation center.



Then round about that place there grew a ledger of forms and requests thicker every cycle, until at last the whole center was hidden from view, and nothing of it could be seen but the tape on the shelf. And a rumor went about in all that Company of the beautiful Waitin Data of the QUEUED T for so was the maiden ASCII data called; and from time to time many devices came and tried to force their way through the ledger, but it was impossible for them to do so, for the forms held fast together like strong passwords, and the young devices were trapped by them, and not being able to go on-hook they idled to a lamentable overload and circuits burned out from early life failure, and so they gave up their efforts from the remote regions.

Many a long cycle afterwards there came a Laser Charge Coupled instruction Device (LCCD)* into that company, and he read an old PROC LIB in Write Only Memory how there should be a CACHE L Center stacking behind the ledger of forms, and that there an enchanted QUEUED T named Waitin Data had NULLA BIDE for MEGA times slices and GIGA cycles and with her the REG/AL unit and the QUEUE$_n$, and the whole foreground. The old PROC LIB had been used by many devices that had sought to pass the ledger of forms, but had been caught and idled by the forms, and died of a miserable budget cut for lack of acceptable performance. Then said the young LCCD, "Nevertheless, I do fear not to try; I shall charge up and break through and see the lovely Waitin Data." The good old PROC LIB and a well planned memory map tried to unbuffer him, but he would not fasten to their bypass words, regardless of their chaining techniques.

For now the MEGA time slices and GIGA cycles were at end, and the NANO second had come when Waitin Data should be activated. When the laser charge coupled device (LCCD) drew heatedly near the ledger of forms, it was changed into a rack of microforms of brilliant and large fiched flowers, whose petals parted and scanned aside to let him pass, and then closed behind him in a thick hedge. When he reached the CACHE L yard, he saw the MACROS and bundled accumulator Regs lying asleep, and on the drum the Pregions were sitting with their R/W heads under their access arms lying idly on the tracks. And when he came indoors, the files on the Vols were ASWAP, the controller in the diskend had his micro IPLD to strike the WCS, and the diskend mount had the pack serial on her lap ready to ISAM. Then he mounted higher, and saw in the hall the whole central processor lying in wait, and above them all, on their PSW's slept the REG/AL unit and the QUEUE$_n$. And still he went farther, and all was so quiet that he could hear his own bufferings and at last he came to the old translator, and went up the winding search argument, and turned the index key in the rusty lock and OP'd the INSTR of the little CRYPTIC MODULE ROM where Sleeping QUEUED T lay. And when he saw her linking so lovely in her wait, he could not turn away his II's; and presently he stacked and charged her (well he did not laser her right away), as he added his error correction code to her parity, and she awakened and OP'd her II's, and linked very kindly on him. And she rose, and they went forth in double precision as the power of his Mantissa floated them past the exponents of SLOTH, and together they awoke the REG/AL unit and the QUEUE$_n$, and the whole central process waked up and linked on each other with great pointers of displacement, and the MACROS in the compiler got up and called themselves, the NOPs sprang up and BALRD their trails, the P regions on the drum drew their heads up from off the tracks, spun round and flew into the fields of the files on the Vols, the diskend files packed up and scanned, and VTOC'd the directory, the JCL on the SYS began to Load, the whole place came to life.

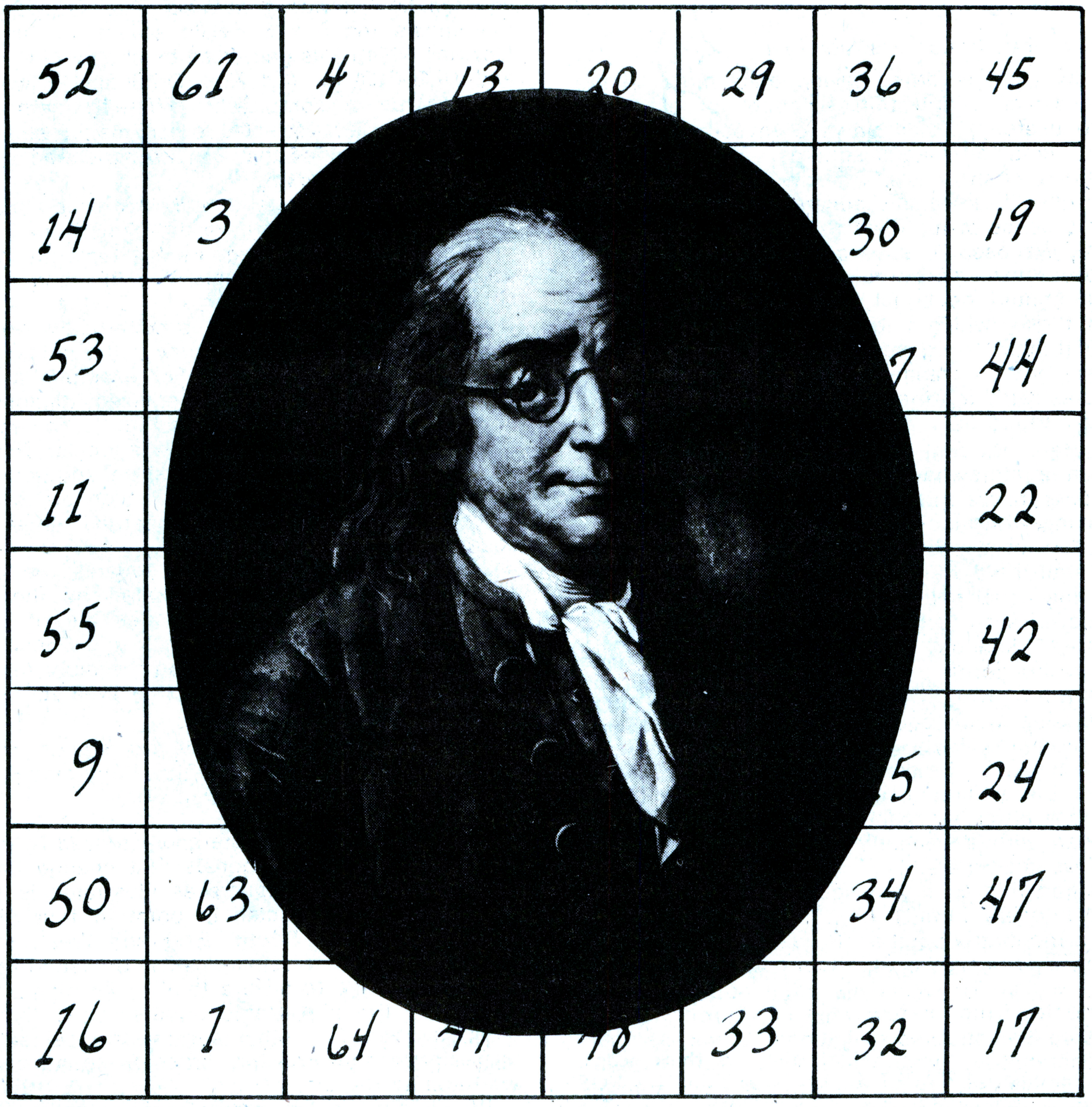
Then the merger of the laser charge coupled device and Maiden ASCII Data was held with all splendor, thus, she was became a High Bride in upper storage (HYBRID DATA as the QUEUE$_n$ had originally listed), and the facilities managed together very happily ever after.

CAST OF CHARACTERS (NO COLLATING SEQUENCE)

REG/AL unit – REGistered Arithmetic Logic Unit
QUEUE$_n$ – QUEUE subset Value N
HYBRID DATA – A coupling of ASCII & EBCDIC
DIOCS's – Special Dock for Loading Instructions
Freg – Formal Register for hopping instructions
IMS elf – Leperchan of Informal Mux Source
HERSelf – Leperchan of HEuRistic Service
Core L'd – Core Load
JOYVIAL – Happy Bubble Memory Container
LCCD – Reminiscent of EL CID – A Hero Type
WOM – Write only Memory – Top Secret Device

He who serves less than full measure at a Hexadecimal Dinner, serves ILL to ALL.

Magic Squares on the Computer


52 61 4 13 20 29 36 45
14 3 30 19
53 44
11 22
55 42
9 24
50 63 34 47
16 1 64 33 32 17

Donald T. Piele
Assistant Professor of Mathematics
University of Wisconsin-Parkside
Kenosha, Wisconsin 53140

Magic squares? Humbug! I've never been able to get excited over someone's special arrangement of numbers that total up to the same sum whether you add across a row or down a column or diagonally. Benjamin Franklin when first confronted with them wrote,

> ". . . it is perhaps a mark of good sense of our (English) mathematicians that they would not spend their time in things that were merely *difficiles nugae* incapable of any useful applications."(1)

Franklin had to confess, however,

> "In my younger days, having once more leisure time (which I still think I might of employed more usefully) I had amused myself in making these kind of magic squares, and, at length acquired such a knack at it, that I could fill the cells of any magic square of reasonable size with a series of numbers as fast as I could write them, disposed in such a manner that the sum of every row, horizontal, perpendicular or diagonal, should be equal; but not being satisfied with these, which I looked on as common and easy things, I imposed on myself more difficult tasks, and succeeded in making other magic squares with a variety of properties, and much more curious."(2)

In spite of the fact that I knew Benjamin Franklin had been a statesman, a scientist, a politician, a philosopher, and a writer, I was surprised to discover that playing with magic squares was also among his lengthy list of avocations. Reading further, I discovered that there are ways of testing magic squares, besides the usual rows, columns, or diagonals, that I had never seen before. For example, there are generalized diagonals, broken diagonals, corner diagonals, horizontal zig-zags, vertical zig-zags, just to name a few. Next, I found that algorithms existed for generating magic squares which looked relatively easy to program. Maybe magic squares aren't so bad after all? Besides, the computer can be programmed to do all the arithmetic and print out a listing of magical properties for each square. That did it.

I began with a generalized version of the algorithm of De la Loubère.(3)
This method fills an n x n square matrix with consecutive integers from 1 to n^2 by putting the ith integer in the matrix position P_i as follows:

1. Place the number 1 in any initial position, $P_i = (i,j)$. The standard initial position is the middle of the top row, $(1,(n+1)/2)$ for n odd.
2. Place the successive integers in vacant cells separated by jumps (A,B), $P_i = P_{i-1} + (A,B)$.
3. If P_i moves outside the square n x n matrix, adjust the coordinates modulo (1,2,3, . . . ,n) so that P_i moves back into the square. e.g. For n = 3, (2,4) = (2,1) and (0,3) = (3,3).
4. If you encounter a position P_j, $j \leq n^2$, that has already been filled, switch for one move to the rule $P_j = P_{j-1} + (C,D)$ and continue as in 2.

De la Loubère's original method specified that 1 be placed in the middle of the top row, $P_1 = (1,(n+1)/2)$, and that (A,B,C,D) be fixed at (1,1,0,—1). This is illustrated in the first sample run of the De la Loubère program. But what happens when you try different starting positions P_1 and other step values (A,B,C,D)? Will any choice of (A,B,C,D) generate a square? De la Loubère used his algorithm only for odd order squares, what happens for even order squares? Given a De la Loubère magic square, can you tell how it was generated? The second sample run shows the 5 x 5 magic square of Backet de Méziriac which was generated by choosing P_1 = (3,4) and (A,B,C,D) = (1,1,2,0). Originally it was constructed by a completely different method (see [1] p. 17). Can you find other magic squares in books or magazines that can be generated with the De la Loubère program?

For an n x n square of numbers to be considered magic it must at least have the same sum for each row and column. If the square is filled with the consecutive numbers 1 through n^2 then each row and column must add up to $n(n^2 + 1)/2$ (why?). All other ways of finding n numbers, symmetrically arranged, that add up to this sum, improves the magic square and makes it more unique. For example, a square may be summed along generalized diagonals as illustrated for a 3 x 3 square in Fig. 1. For an n x n square there are 2n generalized diagonals. The De la Loubère program checks them all in addition to the rows and columns. The best you can do, with this program, is find 4n magical properties for an n x n square.

Benjamin Franklin's magic squares are entirely different and cannot be generated by the De la Loubère algorithm. The best ones are of order 8 and 16, known as the Franklin Magic Squares. The largest one is considered among the most ingenious ever developed. It was impossible even for Franklin to be modest about it.

> ". . . you will readily allow the square of 16 to be the most magically magic of any magic square ever made by any magician."(4)

The Franklin squares are characterized by magical sums along broken diagonals that change direction halfway through the square as illustrated in Fig. 2. They can be constructed to point in four different directions; North, South, East and West. In each direction an n x n square has n broken diagonals, so it is possible to have a total of 4n magic broken diagonals. Franklin Magic Squares have the maximum number. Two other special arrangements that characterize the Franklin order 8 squares are illustrated in Fig. 3.

It is not known how Franklin generated his squares, although it is very likely that they were geometrically motivated. Several investigations have found unique symmetries in the way the numbers are arranged (see [1] p. 93 and [4]). However, there exists an analytical algorithm for reconstructing his squares called the method of *alternation with binate transposition* (see [1] pp. 100-106). It sounds difficult but it is really not. In fact the method can be easily generalized to construct much more than just the Franklin Magic Square. Since there is no difference in the algorithm for squares of order 8 or 16, I will describe, for convenience, the order 8 scheme.

---

1. Originally from *Letters and Papers on Philosophical Subjects* by Benjamin Franklin, LL.D., F.R.S., London, 1769. See [1] p. 89.

2. See footnote 1 pp. 89-90.

3. "De la Loubère was the envoy of Louis XIV to Siam in 1687-1688, and there learnt his method." See [3] p. 195.

4. See footnote 1 p. 93.

Begin with the *plan of construction matrix* (Fig. 4). Number the rows and columns, as usual, and let RC stand for the number at the intersection of row R and column C. The magic square is created as follows:

1. Choose a permutation of the row values, 1 through 8, and denote it by $R_1, R_2, \ldots, R_8$. For the Franklin Square choose 7,8,1,2,3,4,5,6. Let $\bar{R}_i$ be the complementary row $9 - R_i$.
2. Choose an arrangement of the column values 1 through 8, and denote it by C_1, C_2, $\bar{C}_1$, $\bar{C}_2$, C_3, C_4, $\bar{C}_3$, $\bar{C}_4$, such that $C_i + \bar{C}_i = 9$. For the Franklin Square choose 4,6,5,3,7,1,2,8. Notice that 4 + 5, 6 + 3, 7 + 2, and 1 + 8 all equal 9.
3. Rearrange the numbers in the *plan of construction matrix* as shown in Fig. 5. For example, using the row and column sequence given in 1 and 2 for the Franklin Magic Square, $R_1 = 7$ and $C_1 = 4$. Thus, $R_1 C_1 = 52$ and is found in the *plan of construction matrix* at the intersection of row 7 and column 4.

Notice the repetition in C values as you move across the columns and in R values as you move down the rows. This characterizes the Franklin squares and makes the computer algorithm relatively short (see program listing).

The original Franklin Magic Square is generated in the first sample run and has a total of 50 magical sums. It has a few other nice properties too, but they are not tested for here (see [1] p. 96). What happens when you try other permutations? If you ignore the restriction on the column permutations, the computer still generates a square but some numbers will be repeated. Try it! Each new row and column permutation will generate a Franklin-like magic square. But, given a Franklin type magic square, can you find a row and column permutation that will generate it?

It probably never occurred to Franklin that anyone would want to, much less be able to, improve upon his ". . . most magically magic of any magic square." But magic square buffs are a tenacious lot and they should never be underestimated. The most obvious weakness with Franklin's square exists on the main diagonals which are not magic. Many devotees of the subject have tried in vain to remove this imperfection. In 1945, Andrew S. Anema succeeded by constructing, for the first time, a magic square that has all the Franklin properties and in addition is magic along the main diagonals and generalized diagonals (see [2]). His method uses complementary pairs and takes three pages to describe. It turns out that you can generate Anema's improved Franklin square, and many others like it, with the Franklin program described here (see sample run 2). Can you generate other improved Franklin Magic Squares? There are lots of them.

The literature on magic squares is enormous. Probably no other single recreational topic has had more written about it. With very limited experience, my impression is that many of the special methods that have been devised to construct magic squares are merely special cases of more general algorithms. Students who are interested and have a little knowledge of BASIC should be able to step into this area and, with the computer, perform a little magic of their own.

Happy hunting!

POSTSCRIPT

I'm not sure exactly how many magical properties there are in a Franklin Magic Square, but I do know that the number is much larger than I, or Franklin, ever dreamed. This became apparent one evening when I discovered, or perhaps rediscovered, 139 additional magical arrangements already present and waiting to be counted in a Franklin Square of order 8.

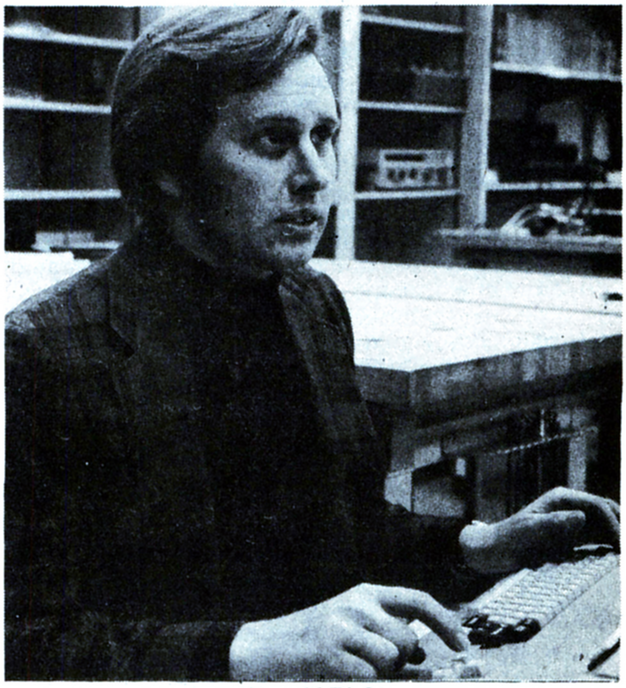
It is a relatively easy exercise to add three subroutines to the Franklin program to check these arrangements for magical sums. Can you do it? Franklin squares appear to have magical properties almost everywhere you look. Can you find other arrangements that sum to 260?

Again, Happy Hunting!

References

1. Andrews, W.S. *Magic Squares And Cubes.* The Open Court Publishing Co. 1908.
2. Anema, Andrew S. "Franklin Magic Squares." *Scripta Mathematica* 11:88-96; 1945.
3. Ball, W.W. Rouse. *Mathematical Recreations and Essays.* The Macmillan Co. New York. 1947.
4. Bragdon C. "The Franklin 16 x 16 Magic Square." *Scripta Mathematica* 4:158-60; 1936.

Photograph: Benjamin Franklin 1706-1790
Oval P.M. Alix, c. 1790, after painting by C.P.A. Van Loo, c. 1777-1785, at the American Philosophical Society, New York.



Donald Piele

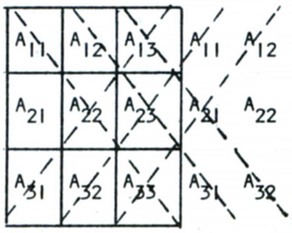


Fig. 1 Generalized diagonals for a 3 x 3 square.

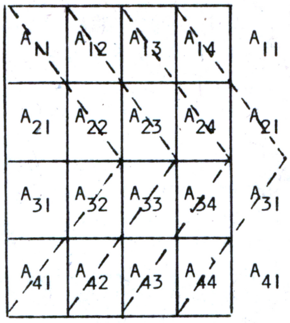


Fig. 2 Broken diagonals in one direction for a 4 x 4 square.

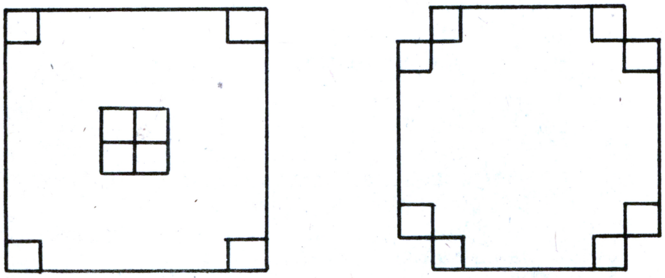


Fig. 3 Special arrangements in the Franklin Magic Square that are magic.

a. Center 4 squares and four corner squares.

b. The 4 near-corner squares.

Columns

ROWS	1	2	3	4	5	6	7	8
1	1	2	3	4	5	6	7	8
2	9	10	11	12	13	14	15	16
3	17	18	19	20	21	22	23	24
4	25	26	27	28	29	30	31	32
5	33	34	35	36	37	38	39	40
6	41	42	43	44	45	46	47	48
7	49	40	51	52	53	54	55	56
8	57	58	59	60	61	62	63	64

Fig. 4 The Plan of Construction Matrix.

R_1C_1	$R_2\bar{C}_1$	R_3C_1	$R_4\bar{C}_1$	R_5C_1	$R_6\bar{C}_1$	R_7C_1	$R_8\bar{C}_1$
$\bar{R}_1C_2$	$\bar{R}_2\bar{C}_2$	$\bar{R}_3C_2$	$\bar{R}_4\bar{C}_2$	$\bar{R}_5C_2$	$\bar{R}_6\bar{C}_2$	$\bar{R}_7C_2$	$\bar{R}_8\bar{C}_2$
$R_1\bar{C}_1$	R_2C_1	$R_3\bar{C}_1$	R_4C_1	$R_5\bar{C}_1$	R_6C_1	$R_7\bar{C}_1$	R_8C_1
$\bar{R}_1\bar{C}_2$	$\bar{R}_2C_2$	$\bar{R}_3\bar{C}_2$	$\bar{R}_4C_2$	$\bar{R}_5\bar{C}_2$	$\bar{R}_6C_2$	$\bar{R}_7\bar{C}_2$	$\bar{R}_8C_2$
R_1C_3	$R_2\bar{C}_3$	R_3C_3	$R_4\bar{C}_3$	R_5C_3	$R_6\bar{C}_3$	R_7C_3	$R_8\bar{C}_3$
$\bar{R}_1C_4$	$\bar{R}_2\bar{C}_4$	$\bar{R}_3C_4$	$\bar{R}_4\bar{C}_4$	$\bar{R}_5C_4$	$\bar{R}_6\bar{C}_4$	$\bar{R}_7C_4$	$\bar{R}_8\bar{C}_4$
$R_1\bar{C}_3$	R_2C_3	$R_3\bar{C}_3$	R_4C_3	$R_5\bar{C}_3$	R_6C_3	$R_7\bar{C}_3$	R_8C_3
$\bar{R}_1\bar{C}_4$	$\bar{R}_2C_4$	$\bar{R}_3\bar{C}_4$	$\bar{R}_4C_4$	$\bar{R}_5\bar{C}_4$	$\bar{R}_6C_4$	$\bar{R}_7\bar{C}_4$	$\bar{R}_8C_4$

Fig. 5 Alternation with permutation scheme.

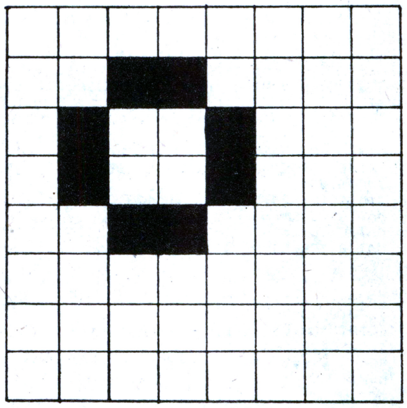


Figure 6. Magic Octagons. Each octagon arrangement sums to 260 wherever it is placed on the 8 x 8 square. There are 25 Magic Octagons.

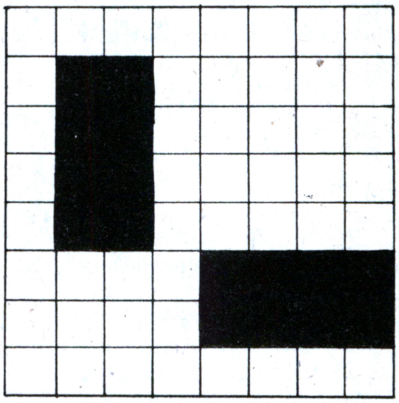


Figure 7. Magic "2 x 4's". Each 2 x 4 rectangle standing up or lying down sums to 260 wherever it is placed on the 8 x 8 square. There are 70 magic "2 x 4's".

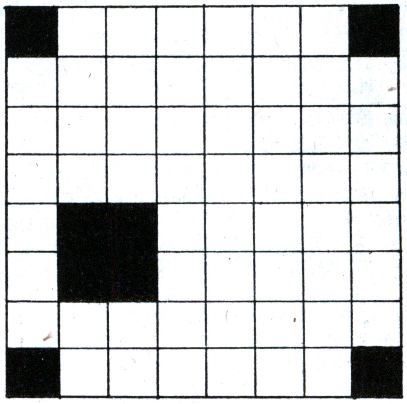


Figure 8. Any 2 x 2 square sums to 130. When it is combined with the four corner squares the sum is 260. There are 45 such arrangements but one has already been counted.

DE LA LOUBERE PROGRAM

```
10   REM PROGRAM WRITTEN BY D.T.PIELE 7/15/75
20   REM      ******  THE DE LA LOUBERE PROGRAM   ******
30   PRINT "THIS PROGRAM TESTS NXN SQUARES OF ODD ORDER FOR "
40   PRINT "THEIR MAGICAL PROPERTIES. YOU CAN EITHER ENTER YOUR"
50   PRINT "OWN SQUARE OR LET THE COMPUTER GENERATE ONE FOR YOU "
60   PRINT "USING THE ALGORITHM OF DE LA LOUBERE."
70   PRINT "FOR THE COMPUTER SQUARE TYPE 1. FOR YOUR OWN TYPE 2."
80   INPUT X
90   PRINT "HOW MANY ROWS DO YOU WANT?"
100   PRINT "PICK AND ODD NUMBER BETWEEN 3 AND 11."
110   INPUT N
120   DIM A[12,12],C[12,12],B[12]
130   MAT A=ZER[N,N]
140   MAT C=ZER[N,N]
150   MAT B=ZER[N]
160   IF X=2 THEN 1190
170   K=0
180   PRINT "PICK THE POSITION I,J FOR 1. (MIDDLE OF ROW 1 IS STANDARD)"
190   INPUT I,J
200   PRINT "CHOOSE A,B,C,D (1,1,0,-1 IS STANDARD)"
210   INPUT A,B,C,D
220   REM   ****** THE ALGORITHM FOR GENERATING THE SQUARE ******
230   K=K+1
240   A[I,J]=K
250   IF K=INT(N*N+.5) THEN 370
260   I=I-B
270   J=J+A
280   IF I=0 THEN 700
290   IF I<0 THEN 740
300   IF I>N THEN 700
310   IF J=0 THEN 720
320   IF J<0 THEN 760
330   IF J>N THEN 720
340   IF A[I,J]>0 THEN 670
350   GOTO 230
360   REM ******  END OF ALGORITHM  ******
370   PRINT "HERE IS YOUR SQUARE OF ORDER"N
380   PRINT
390   PRINT
400   MAT  PRINT A;
410   S=INT(N*(N*N+1)/2+.5)
420   MAT C=A
430   GOSUB 790
440   R1=C
450   MAT C=TRN(A)
460   GOSUB 790
470   C1=C
480   PRINT
490   PRINT "HERE IS A LIST OF ITS MAGICAL PROPERTIES."
500   PRINT
510   PRINT "ROWS AND COLUMNS:"R1+C1
520   GOSUB 920
530   D1=C+D
540   PRINT "GENERALIZED DIAGONALS:"D1
550   PRINT
560   PRINT "TOTAL MAGICAL SUMS: "R1+C1+D1
570   PRINT
580   PRINT "DO YOU WISH TO TRY AGAIN? TYPE 1 FOR YES, 0 FOR NO."
590   INPUT Y
600   PRINT
610   IF Y=1 THEN 70
620   PRINT "GOODBYE. SEE YOU AT THE FRANKLIN FESTIVAL"
630  PRINT "OCT. 5 TO 11 AT UW-PARKSIDE."
640  PRINT
650  PRINT
660  STOP
670  I=I+B-D
680  J=J-A+C
690  GOTO 280
700  I=ABS(I-N)
710  GOTO 290
720  J=ABS(J-N)
730  GOTO 320
740  I=N+I
750  GOTO 300
760  J=J+N
770  GOTO 330
780  REM  ****** TEST ROWS AND COLUMNS ******
790  C=0
800  FOR I=1 TO N
810  E=0
820  FOR J=1 TO N
830  E=E+C[I,J]
840  NEXT J
850  IF E=S THEN 880
860  GOTO 890
870  GOTO 890
880  C=C+1
890  NEXT I
900  RETURN
910  REM ****** TEST GENERALIZED DIAGONALS ******
920  C=0
930  D=0
940  FOR J=0 TO N-1
950  E=0
960  F=0
970  FOR I=1 TO N
980  R=I+J
990  T=N+1-I-J
1000  IF R <= N THEN 1060
1010  R=ABS(R-N)
1020  GOTO 1060
1030  IF T >= 1 THEN 1080
1040  T=T+N
1050  GOTO 1080
1060  E=E+A[I,R]
1070  GOTO 1030
1080  F=F+A[T,I]
1090  NEXT I
1100  IF E=S THEN 1130
1110  IF F=S THEN 1150
1120  GOTO 1160
1130  C=C+1
1140  GOTO 1110
1150  D=D+1
1160  NEXT J
1170  RETURN
1180  REM  ****** ENTER YOUR OWN SQUARE ******
1190  PRINT "LIST THE MEMBERS OF EACH ROW SEPARATED BY A COMMA."
1200  FOR R=1 TO N
1210  PRINT "ROW"R
1220  MAT  INPUT B
1230  FOR I=1 TO N
1240  A[R,I]=B[I]
1250  NEXT I
1260  PRINT
1270  NEXT R
1280  PRINT
1290  GOTO 370
1300  END
```

SAMPLE RUN

1

```
THIS PROGRAM TESTS NXN SQUARES OF ODD ORDER FOR
THEIR MAGICAL PROPERTIES. YOU CAN EITHER ENTER YOUR
OWN SQUARE OR LET THE COMPUTER GENERATE ONE FOR YOU
USING THE ALGORITHM OF DE LA LOUBERE.
FOR THE COMPUTER SQUARE TYPE 1. FOR YOUR OWN TYPE 2.
?1
HOW MANY ROWS DO YOU WANT?
PICK AND ODD NUMBER BETWEEN 3 AND 11.
?3
PICK THE POSITION I,J FOR 1. (MIDDLE OF ROW 1 IS STANDARD)
?1,2
CHOOSE A,B,C,D (1,1,0,-1 IS STANDARD)
?1,1,0,-1
HERE IS YOUR SQUARE OF ORDER 3


 8     1     6

 3     5     7

 4     9     2



HERE IS A LIST OF ITS MAGICAL PROPERTIES.

ROWS AND COLUMNS: 6
GENERALIZED DIAGONALS: 2

TOTAL MAGICAL SUMS:  8

DO YOU WISH TO TRY AGAIN? TYPE 1 FOR YES, 0 FOR NO.
?1
```

2

```
FOR THE COMPUTER SQUARE TYPE 1. FOR YOUR OWN TYPE 2.
?1
HOW MANY ROWS DO YOU WANT?
PICK AND ODD NUMBER BETWEEN 3 AND 11.
?5
PICK THE POSITION I,J FOR 1. (MIDDLE OF ROW 1 IS STANDARD)
?3,4
CHOOSE A,B,C,D (1,1,0,-1 IS STANDARD)
?1,1,2,0
HERE IS YOUR SQUARE OF ORDER 5


 3     16     9     22     15

 20    8      21    14     2

 7     25     13    1      19

 24    12     5     18     6

 11    4      17    10     23


HERE IS A LIST OF ITS MAGICAL PROPERTIES.

ROWS AND COLUMNS: 10
GENERALIZED DIAGONALS: 2

TOTAL MAGICAL SUMS:  12

DO YOU WISH TO TRY AGAIN? TYPE 1 FOR YES, 0 FOR NO.
?0

GOODBYE. SEE YOU AT THE FRANKLIN FESTIVAL
OCT. 5 TO 11 AT UW-PARKSIDE.
```

FRANKLIN PROGRAM

```
10  REM PROGRAM WRITTEN BY D.T.PIELE 7/15/75
20  PRINT "THIS IS THE FRANKLIN MAGIC SQUARE PROGRAM."
30  PRINT "IT WILL GENERATE AND TEST 8X8 SQUARES. YOU "
40  PRINT "CAN ALSO ENTER AND TEST YOUR OWN 8X8 SQUARES."
50  PRINT
60  DIM A[8,8],C[8],G[8],D[8,8],B[8],E[8,8],F[8]
70  MAT  READ A,C,G
80  PRINT "FOR THE COMPUTER GENERATED SQAURE TYPE 1."
90  PRINT "TO ENTER YOUR OWN TYPE 2."
100  PRINT
110  INPUT A
120  IF A=2 THEN 1860
130  PRINT "FIRST PERMUTE THE ROWS 1 THROUGH 8 AND SEPARATE"
140  PRINT "WITH COMMAS. FOR THE FRANKLIN SQUARE CHOOSE"
150  PRINT " 7,8,1,2,3,4,5,6"
160  PRINT
170  MAT  INPUT F
180  PRINT
190  PRINT "NEXT SUPPLY A PERMUTATION OF THE COLUMNS 1 THROUGH 8"
200  PRINT "IN THE FORM A,B,C,D,E,F,G,H SUCH THAT A+C=9, B+D=9,"
210  PRINT "E+G=9, AND F+H=9.FOR THE FRANKLIN MAGIC SQUARE CHOOSE"
220  PRINT " 4,6,5,3,7,1,2,8"
230  PRINT
240  MAT  INPUT B
250  REM  ****** ALGORITHM TO GENERATE SQUARES  ******
260  J=1
270  FOR N=1 TO 7 STEP 2
280  FOR I=1 TO 7 STEP 2
290  K=I+1
300  R=B[I]
310  L=F[N]
320  S=B[K]
330  M=9-L
340  D[I,J]=A[L,R]
350  D[K,J]=A[M,S]
360  NEXT I
370  J=J+2
380  NEXT N
390  J=2
400  FOR N=2 TO 8 STEP 2
410  FOR I=1 TO 7 STEP 2
420  K=I+1
430  R=9-B[I]
440  S=9-B[K]
450  L=F[N]
460  M=9-L
470  D[I,J]=A[L,R]
480  D[K,J]=A[M,S]
490  NEXT I
500  J=J+2
510  NEXT N
520  PRINT
530  PRINT "YOUR 8X8 SQUARE IS"
540  PRINT
550  MAT  PRINT D;
560  REM  ****** END OF THE ALGORITHM ******
570  PRINT
580  FOR I=1 TO 8
590  IF F[I] <> G[I] THEN 670
600  NEXT I
610  FOR I=1 TO 8
620  IF B[I] <> C[I] THEN 670
630  NEXT I
640  PRINT " THIS IS THE BENJAMIN FRANKLIN MAGIC SQUARE OF ORDER 8."
650  PRINT
660  REM  ****** TABULATION OF THE MAGICAL PROPERTIES ******
670  PRINT "HERE IS A LIST OF ITS MAGICAL PROPERTIES."
680  PRINT
690  MAT E=D
700  GOSUB 1420
710  R1=C
720  MAT E=TRN(D)
730  GOSUB 1420
740  C1=C
750  PRINT "ROWS AND COLUMNS:"R1+C1
760  M1=0
770  M=D[1,1]+D[2,2]+D[3,3]+D[4,4]+D[5,5]+D[6,6]+D[7,7]+D[8,8]
780  IF M <> 260 THEN 800
790  M1=1
800  M=D[8,1]+D[7,2]+D[6,3]+D[5,4]+D[4,5]+D[3,6]+D[2,7]+D[1,8]
810  IF M <> 260 THEN 830
820  M1=M1+1
830  PRINT "MAIN DIAGONALS:"M1
840  G=0
850  FOR J=1 TO 7
860  E=0
870  F=0
880  FOR I=1 TO 8
890  R=I+J
900  T=9-I-J
910  IF R <= 8 THEN 930
920  R=R-8
930  E=E+D[I,R]
940  IF T >= 1 THEN 960
950  T=T+8
960  F=F+D[T,I]
970  NEXT I
980  IF E <> 260 THEN 1000
990  G=G+1
1000  IF F <> 260 THEN 1020
1010  G=G+1
1020  NEXT J
1030  PRINT "GENERALIZED DIAGONALS:"G
1040  MAT E=D
1050  GOSUB 1520
1060  B1=C
1070  MAT E=TRN(D)
1080  GOSUB 1520
1090  B2=C
1100  MAT E=D
1110  GOSUB 1660
1120  B3=C
1130  MAT E=TRN(D)
1140  GOSUB 1660
1150  Y=B1+B2+B3+C
1160  PRINT "BROKEN DIAGONALS:"Y
1170  C=0
1180  D=0
1190  E=D[1,1]+D[1,8]+D[8,1]+D[8,8]+D[4,4]+D[5,4]+D[4,5]+D[5,5]
1200  C=1
1210  PRINT "SPECIAL CASES:"
1220  PRINT "           CENTER FOUR SQUARES PLUS FOUR CORNER SQUARES."
1230  E=D[1,2]+D[2,1]+D[1,7]+D[2,8]+D[7,1]+D[7,8]+D[8,2]+D[8,7]
1240  IF E <> 260 THEN 1270
1250  PRINT "           THE FOUR CORNER DIAGONAL PAIRS."
1260  D=1
1270  PRINT
1280  W=R1+C1+Y+C+D+M1+G
1290  PRINT "TOTAL MAGICAL SUMS:"W
1300  PRINT
1310  PRINT "DO YOU WANT TO TRY AGAIN? TYPE 1 FOR THE COMPUTER SQUARE,"
1320  PRINT "TYPE 2 TO ENTER YOUR OWN SQUARE, AND TYPE 0 TO STOP."
1330  PRINT
1340  INPUT Z
1350  IF Z=1 THEN 130
1360  IF Z=2 THEN 1860
1370  PRINT "GOODBYE. SEE YOU AT THE BENJAMIN FRANKLIN FESTIVAL"
1380  PRINT "AT UW-PARKSIDE, OCTOBER 5 TO 11."
1390  PRINT
1400  STOP
1410  REM  ****** SUBROUTINES TO CHECK FOR MAGICAL SUMS ******
1420  C=0
1430  FOR I=1 TO 8
1440  E=0
1450  FOR J=1 TO 8
1460  E=E+E[I,J]
1470  NEXT J
1480  IF E <> 260 THEN 1500
1490  C=C+1
1500  NEXT I
1510  RETURN
1520  C=0
1530  FOR J=0 TO 7
1540  E=0
1550  FOR I=1 TO 4
1560  R=I+J
1570  T=9-I
1580  IF R <= 8 THEN 1600
1590  R=R-8
1600  E=E+E[I,R]+E[T,R]
1610  NEXT I
1620  IF E <> 260 THEN 1640
1630  C=C+1
1640  NEXT J
1650  RETURN
1660  C=0
1670  FOR J=0 TO 7
1680  E=0
1690  FOR I=1 TO 4
1700  R=9-I
1710  T=9-I-J
1720  IF T >= 1 THEN 1740
1730  T=T+8
1740  E=E+E[T,I]+E[T,R]
1750  NEXT I
1760  IF E <> 260 THEN 1780
1770  C=C+1
1780  NEXT J
1790  RETURN
1800  DATA 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16
1810  DATA 17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32
1820  DATA 33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48
1830  DATA 49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64
1840  DATA 4,6,5,3,7,1,2,8,7,8,1,2,3,4,5,6
1850  REM  ****** ENTER YOUR OWN SQUARE ******
1860  PRINT
1870  PRINT "LIST THE MEMBERS OF EACH ROW SEPARATED BY COMMAS."
1880  FOR R=1 TO 8
1890  PRINT "ROW"R
1900  PRINT
1910  MAT  INPUT B
1920  FOR I=1 TO 8
1930  D[R,I]=B[I]
1940  NEXT I
1950  PRINT
1960  NEXT R
1970  GOTO 650
1980  END
```

SAMPLE RUN

1

```
THIS IS THE FRANKLIN MAGIC SQUARE PROGRAM.
IT WILL GENERATE AND TEST 8X8 SQUARES. YOU
CAN ALSO ENTER AND TEST YOUR OWN 8X8 SQUARES.

FOR THE COMPUTER GENERATED SQAURE TYPE 1.
TO ENTER YOUR OWN TYPE 2.

?1
FIRST PERMUTE THE ROWS 1 THROUGH 8 AND SEPARATE
WITH COMMAS. FOR THE FRANKLIN SQUARE CHOOSE
 7,8,1,2,3,4,5,6

?7,8,1,2,3,4,5,6

NEXT SUPPLY A PERMUTATION OF THE COLUMNS 1 THROUGH 8
IN THE FORM A,B,C,D,E,F,G,H SUCH THAT A+C=9, B+D=9,
E+G=9, AND F+H=9.FOR THE FRANKLIN MAGIC SQUARE CHOOSE
 4,6,5,3,7,1,2,8

?4,6,5,3,7,1,2,8

YOUR 8X8 SQUARE IS

 52     61     4      13     20     29     36     45

 14     3      62     51     46     35     30     19

 53     60     5      12     21     28     37     44

 11     6      59     54     43     38     27     22

 55     58     7      10     23     26     39     42

 9      8      57     56     41     40     25     24

 50     63     2      15     18     31     34     47

 16     1      64     49     48     33     32     17


 THIS IS THE BENJAMIN FRANKLIN MAGIC SQUARE OF ORDER 8.

HERE IS A LIST OF ITS MAGICAL PROPERTIES.

ROWS AND COLUMNS: 16
MAIN DIAGONALS: 0
GENERALIZED DIAGONALS: 0
BROKEN DIAGONALS: 32
SPECIAL CASES:
          CENTER FOUR SQUARES PLUS FOUR CORNER SQUARES.
          THE FOUR CORNER DIAGONAL PAIRS.

TOTAL MAGICAL SUMS: 50

DO YOU WANT TO TRY AGAIN? TYPE 1 FOR THE COMPUTER SQUARE,
TYPE 2 TO ENTER YOUR OWN SQUARE, AND TYPE 0 TO STOP.
```

2

```
?1
FIRST PERMUTE THE ROWS 1 THROUGH 8 AND SEPARATE
WITH COMMAS. FOR THE FRANKLIN SQUARE CHOOSE
 7,8,1,2,3,4,5,6

?1,2,8,7,3,4,6,5

NEXT SUPPLY A PERMUTATION OF THE COLUMNS 1 THROUGH 8
IN THE FORM A,B,C,D,E,F,G,H SUCH THAT A+C=9, B+D=9,
E+G=9, AND F+H=9.FOR THE FRANKLIN MAGIC SQUARE CHOOSE
 4,6,5,3,7,1,2,8

?1,2,8,7,5,6,4,3

YOUR 8X8 SQUARE IS

 1      16     57     56     17     32     41     40

 58     55     2      15     42     39     18     31

 8      9      64     49     24     25     48     33

 63     50     7      10     47     34     23     26

 5      12     61     52     21     28     45     36

 62     51     6      11     46     35     22     27

 4      13     60     53     20     29     44     37

 59     54     3      14     43     38     19     30

HERE IS A LIST OF ITS MAGICAL PROPERTIES.

ROWS AND COLUMNS: 16
MAIN DIAGONALS: 2
GENERALIZED DIAGONALS: 14
BROKEN DIAGONALS: 32
SPECIAL CASES:
          CENTER FOUR SQUARES PLUS FOUR CORNER SQUARES.
          THE FOUR CORNER DIAGONAL PAIRS.

TOTAL MAGICAL SUMS: 66

DO YOU WANT TO TRY AGAIN? TYPE 1 FOR THE COMPUTER SQUARE,
TYPE 2 TO ENTER YOUR OWN SQUARE, AND TYPE 0 TO STOP.

?0
GOODBYE. SEE YOU AT THE BENJAMIN FRANKLIN FESTIVAL
AT UW-PARKSIDE, OCTOBER 5 TO 11.
```

More Input/Output

REJOINDER !

Dear Editor:

Thank you for publishing my complaint letter in the March-April 1975 issue and the letter by "name withheld by request" in the September-October 1975 issue.

May I first advise my colleague that I do *not* in any way control even nominally any of the computer education programs in our city. Nor would I wish to.

I am naturally concerned with the uses and misuses of calculators and computers used in *mathematics* education. And there is a great misuse not only in New York but all over the country!

In my note I define *Computer Mathematics* as "an enrichment program where the computer is used to motivate students to search for problem areas requiring computers for solution or exploration." When I taught Computer Mathematics courses in high schools I introduced my students to topics in number theory, statistics, numerical integration, series evaluation, linear algebra, matrix operation, and more. It was my hope that by opening the eye of students to the vast "world of mathematics" they would continue to search, read, and learn—an open-ended approach.

But to see computers used to evaluate simple formulas and to "prove" the commutativity of multiplication makes me wish that teachers would take workshops to extend their own horizons.

To see plotters attached to programmable calculators used to vary a and b in plotting y = a sin bx is another waste of computer- and student-time.

"Computer Mathematics" is only one way in which computers can be used in education. I hope *Creative Computing* will serve to introduce readers to creative ways which do not lead to dead-end use of computers by students who can do so much more.

Priorities should be established by educators who have the use of computers for instructional purposes. For enrichment purposes I recommend "Computer Mathematics" as defined above. It has other uses for other students in varied subject areas.

George Grossman
Director of Mathematics
Board of Education of the City of New York

NOTORIOUS 196 !

Dear Editor:

I have run a palindrome-finding program on the notorious number 196 and wish to warn others of the negative results they can expect on it.

After 79,098 iterations of transpose-and-add, no palindrome yet. The number had well over 31,000 digits. The run took about 3 hours on a CDC 6600.

I have only run one test, but the digits 0-9 do *not* appear to be uniformly distributed by this process. I let the program run 4,780 cycles and counted digits. As extremes, there were 468,042 sixes and 498,231 nines, out of 4,787,887 total digits. These extremes seem too large to be categorized as uniform. (Complementary digits have equal frequencies; e.g. there were 468,047 threes.)

I hope this will spare someone the expense of running their computer all night.

Lynn D. Yarbrough
Lexington, Mass

The Computer ''Glass Box''

Teaching With *A Programming Language*

Howard A. Peelle
University of Massachusetts

Introduction

The COMPUTER GLASS BOX is a bold new approach to teaching with A Programming Language.[1] In this approach, short and quickly comprehensible computer programs are given to students for their direct viewing. Each program embodies a concept, a procedure, or a relationship and is written as simply and clearly as possible. The inner workings of such a program are visible and, hence, become the basis for learning.

This approach utilizes a computer program more as a "glass box" than a black box. The program's formal definition – expressed in the explicit terms of a programming language – serves to elucidate and reveal understanding. By observing the structure of a program as well as its behavior, key concepts may become *transparent* to the student.

Related Research

The glass box approach represents a synthesis of ideas put forth by three other researchers. MIT's Seymour Papert has recommended that children study procedures actively by using a computer programming language (called LOGO) as a conceptual framework [1]. Kenneth Iverson of IBM has persistently stressed simplicity and generality in using APL to expose fundamentals in a variety of mathematical and scientific disciplines [2]. IBM's Paul Berry first advocated open use of APL as a strategy for teaching in what he called the "functional approach" [3].

Characteristics of the COMPUTER GLASS BOX Approach

In contrast to conventional computer-assisted instruction (CAI), the glass box approach allows the student significant *control* over his own learning processes. This control is achieved through the activity of programming. Programs can be entered independently by the student via a computer terminal, and their use requires no other prestored curriculum material – as do most CAI applications. Indeed, making the full power of the computer accessible to the learner is 180° from the kind of CAI characterized by programmed instruction, tutorial, or drill-and-test sequences.

This approach is pedagogically suitable for a wide range of educational levels – from elementary school children to university graduate students. Especially for children who have been held powerless in lock-step educational systems, use of the computer in this way opens up new worlds of learning – *active* learning, learning with *power.*

Using glass box computer programs, students can proceed to learn during several complementary activities. Specifically, they can:

examine the program's definition (intuitively)
analyze the program's definition (logically)
predict the outcomes of the program
execute the program on a computer
scrutinize the program's behavior
experiment with different applications of the program
modify or expand the program
generalize the program
invent new or related programs, and
discuss implications with teachers and peers.

These student-initiated, student-responsible, success-oriented activities differ dramatically from frantic hand-waving about abstract concepts often seen in classrooms.

The ideal glass box program is also expository – it 'speaks' to its reader, explicating concepts and procedures in concrete terms. Desirable characteristics of such a program are:

Simplicity
Comprehensibility
Flexibility
Generality
Elegance
Provocative Implications

By "simplicity" I mean that a single idea of modest scope is to be taught using a brief program (about 10 lines of APL coding, taking less than 5 minutes to type). By "comprehensibility", I mean using clear, readable commands (usually one per line) with well-chosen mnemonic identifiers. By "flexibility" I mean a program design which is easily modified and which can be used with other programs in modular structuring (nested sub-programs with explicit resultants). By "generality" I mean developing mathematical models which can extend to a class of cases. By "elegance" I mean choosing expressions which strike one's aesthetic chords. And, finally, a glass box program is "provocative" when its implications suggest interesting follow-up discussions.

To the extent that these characteristics foster insight and learning, a glass box program is, itself, a *pedagogical agent.*

Examples of Glass Box Programs

To illustrate this approach, some sample glass box APL programs are described below, with accompanying suggestions for extending their use in teaching-learning settings.

COMPUTER – ASSISTED INSTRUCTION

In order to emphasize the contrast with conventional uses of computers for teaching, the first glass box program illustrated is from the area of computer-assisted instruction. Instead of concealing the CAI program – usually designed to control the child's behavior – we show him the mechanism itself so that he may see how it works and ultimately *control the computer.*

Consider the APL program below which exposes the essence of drill-and-practice in multiplication skills. In drill-and-practice, typically, a student is given a series of problems to solve, is asked for his answers, and the answers are judged for correctness, etc. Indeed, the computer is an excellent vehicle for administering drill-and-practice, but a programming language can also *describe* this process clearly.

The DRILL program begins with a NEWPROBLEM and prints 'MULTIPLY', a simplified message telling the student what to do with the two numbers that will follow. The FIRST number is an integer randomly chosen between 1 and 20, and the SECOND number likewise.

---

[1] *A Programming Language* (abbreviated APL) is a new multi-purpose computer programming language developed by Kenneth Iverson of IBM. Originally conceived as a unifying mathematical notation, APL has since been used successfully in fields such as business, scientific research and education.

Again, this suggests an analogy with human behavior: experiencing the most emotion-packed events first and then tapering off may be more tolerable than the reverse.

Other variations of input also suggest interpretation in terms of human psychology. Sandwiching a low-emotion event between two high-emotion events, say 7 2 7, can make the total sequence tolerable; by contrast, the events 7 7 2 and 2 7 7 produce mad behavior.

The mathematics underlying this TEMPER model can be exposed quickly and naturally. For example, after some experimentation with the program, one might wonder: How many 5s can the program take before it 'blows its top?'

```
          TEMPER
     ⎕:                         ⎕:
          5                          5
     ⎕:                         ⎕:
          5                          5
     ⎕:                         ⎕:
          5                          .
     ⎕:                              .
          5                          .
```

A sequence of 5s builds up EMOTION to higher and higher values, but never reaches 10. This process parallels the well-known geometric series 1 1/2 1/4 1/8 1/16 1/32 . . . the sum of which converges to 2. Exploring in this way, a child may gain some insight into the nature of infinite series in an active and interesting (at least less abstract) setting.

Some simple modifications of the TEMPER program students might make are to: (a) change the threshold, e.g. from 10 to 25 for higher tolerance, or to ?25 (a random number) for unpredictable behavior; (b) modify the model, e.g. from EMOTION ÷ 2 to EMOTION ÷ 3 to express stronger 'forgetting'; (c) adapt the program for use by others, e.g. inserting conversational statements such as 'ENTER NUMBERS FROM 1 TO 9' or even 'CAUTION! THIS PROGRAM MAY BECOME EMOTIONAL. . .', and (d) make the program dynamic, e.g. automatically resetting EMOTION to 0 after an emotional catharthis.

Possible extensions of TEMPER include: (a) writing related programs, such as a version with multiple emotional dimensions like ANGER, FEAR, and LOVE, and (b) writing companion programs, such as two TEMPER-like programs which interact with each other so that one's output is the other's input.

CYBERNETICS

In the area of cybernetics, students can be introduced to some sophisticated ideas by using simple computer programs. Scene analysis, for example, is an important part of robotics research. In designing vision machines, it is important to know what types of scenes can be computationally distinguished. Consider the two scenes below:

```
          SCENES


     *********                 *********
     *                         *       *
     *                         *       *
     *                         *       *
     *********                 *********
             *
             *
             *
     *********                 *********
```

One scene is "connected"; the other is not connected. Note that the same line segments comprise the two scenes, but that they are in different positions.

Suppose one of these two SCENES is PICKed at random. Call it MYSTERY.

```
          MYSTERY ← PICK  SCENES
```

Further suppose that you are permitted to PEEK at small portions of the MYSTERY scene -- called "microscenes" -- but you are not told *where* the microscenes came from. For example,

```
          PEEK  MYSTERY

     ****
     *
     *

          PEEK  MYSTERY


     ****


          PEEK  MYSTERY

      *
      ***
```

PEEKing at MYSTERY is like using a flashlight to illuminate small unidentifiable places on a much larger unknown scene.

After a period of probing, the question arises: *Can* you determine which scene it is that you are looking at?[1] (The answer is postponed so that the reader may ponder this question.)

The APL programs which facilitate exploration of this question in scene analysis are simple indeed:

This program will PICK one of 2 SCENES at random for the result called MYSTERY.

```
      ∇ MYSTERY ← PICK  SCENES

  [1]   MYSTERY ← SCENES[?2;;]

      ∇
```

This program will PEEK at some two-dimensional SCENE and produce a random 4 by 4 portion for the result called MICROSCENE.

```
      ∇ MICROSCENE ← PEEK  SCENE

  [1]   MICROSCENE ← SCENE[(⍳4)+(?10);(⍳4)+(?10)]

      ∇
```

The enterprising student might elect to automate the production of MICROSCENEs.

```
          AUTOPEEK                        ∇ AUTOPEEK

                                      [1]   ''

     ***                              [2]   PEEK  MYSTERY
       *
                                      [3]   ''

      ***                             [4]   →1
      *
      *                                   ∇
      *


     *
     *
     ****


     **
      *
      *
      *



        **


      .
      .
      .
```

Soon it should become clear that these two SCENES *cannot* be distinguished on the basis of random microscenes

---

[1] This question is treated as a theorem by Minsky and Papert in their book *Perceptrons*, MIT Press, 1970.

Perceptrons are theoretical machines which can be trained to detect features of a scene by computations in a layered network of logical elements.

```
      ∇ DRILL
[1]     NEWPROBLEM:
[2]     'MULTIPLY'
[3]     ⎕ ← FIRST ← ?20
[4]     ⎕ ← SECOND ← ?20
[5]     ENTER: ANSWER ← ⎕
[6]     →NEWPROBLEM IF ANSWER=FIRST×SECOND
[7]     'NOPE. TRY AGAIN.'
[8]     →ENTER
      ∇
```

The student may ENTER his ANSWER which is then judged for correctness by the program. IF the ANSWER equals the FIRST number times the SECOND number, a NEWPROBLEM is given; otherwise (if ANSWER is wrong) 'NOPE. TRY AGAIN.' is printed, and the student may ENTER his answer again.

NOTE: IF is a sub-program used to facilitate the reading of branching commands. Its definition is:

```
      ∇ BRANCH ← LINE  IF  CONDITION
[1]     BRANCH ← CONDITION/LINE
      ∇
```

Its syntax is → (line number) IF (condition)

It means that IF the condition is true (evaluates to 1), the program branches to the line number (or line label) given; IF the condition is false (evaluates to 0), the program branches to the next line.

In order to use the DRILL program, its name is typed. The following is a sample:

```
      DRILL
MULTIPLY
 19
 2
⎕:
      38
MULTIPLY
 16
 18
⎕:
      248
NOPE. TRY AGAIN.
⎕:
      288
```

```
MULTIPLY
 8
 12
⎕:
      96
MULTIPLY
 6
 2
⎕:
      12
MULTIPLY
 14
 18
⎕:
```

Students notice immediately that this program has a flaw. It does not stop! Scrutinizing the program's definition reveals that after getting a multiplication problem correct, one always gets a new problem — ad infinitum. Also, after getting a problem wrong, the student must answer that same problem again — another potentially endless loop. The student's first task, then, might be to build in an option to stop the program at will.

DRILL is, of course, only a *prototype* program. With other modifications of one's choosing, DRILL may become considerably more sophisticated. Possible extensions include: (a) displaying pictorial feedback — like a "smiley face" for positive reinforcement or a "grouchy face" instead of 'NOPE. TRY AGAIN.', (b) presenting a pre-specified total number of problems, (c) limiting the number of allowable mistakes on individual problems (or all problems), (d) generalizing the multiplicands to create a more flexible range of problems (including negative numbers, decimals, etc.), (e) gathering performance data, (f) using performance criteria to make diagnoses, (g) automatically adapting level of difficulty based on diagnoses, (h) adding personalized instructions, and (i) building in timing components, jump-ahead options and hints.

```
 *****              *****
* o o *            * × × *
*  ∇  *            *  ∆  *
* \_/ *            * /¯\ *
 *****              *****
```

PSYCHOLOGY

With computer programs suitable for viewing, students may learn some fundamentals of psychology. In studying behavior, for example, consider the following APL program[1] which models — albeit crudely — an emotional reaction.

```
      ∇ TEMPER
[1]     EMOTION ← 0
[2]     NEW: EMOTION ← ⎕ + EMOTION÷2
[3]     →MAD IF EMOTION>10
[4]     →NEW
[5]     MAD: '**!?!**?!!'
      ∇
```

TEMPER is a program which will, under certain conditions, 'get mad at you'.

The program begins with zero EMOTION and then encounters a series of numbers, representing 'events' in the life of the program. A low number is low in emotional significance; whereas high numbers are highly emotion-producing.

Each time a number is entered, the program generates a NEW EMOTION based on a simple mathematical model: EMOTION becomes the number just entered plus one half of the previous EMOTION. (In the course of human events, this might be akin to the ameliorating effect of time on emotional burdens, i.e. 'sleeping on your troubles'.)

This process continues until a test condition — the "threshold" for mad behavior — is exceeded. The program goes MAD if EMOTION ever becomes greater than 10. (**!?!**?!! is the computer's programmed vernacular.)

To use the program, the child types its name (TEMPER) and then enters a sequence of numbers. For example:

```
      TEMPER
⎕:
      4
⎕:
      6
⎕:
      8
**!?!**?!!
```

This sequence produced MAD behavior.

But, suppose one tries entering the same numbers in a different order:

```
      TEMPER
⎕:
      8
⎕:
      6
⎕:
      4
⎕:
      →
```

Here, the program does not display MAD: '**!?!**?!!'. Apparently, (for this model) the sequence 8 6 4 is "tolerable," whereas the previous sequence 4 6 8 clearly was not tolerable!

---

[1] This program is similar to one written in a simplified FORTRAN by John Loehlin in *Computer Models of Personality*, Random House, NY, 1968.

alone. (Of course, if one could trace sequentially through a scene, its "connectedness" or "non-connectedness" could be determined easily.)

Possible extensions of this excursion into scene analysis include studying perceptrons and related questions about "spatially local evidence."

COMPUTER ART

The world of computer art can be opened to students through a few simple APL programs. Beginning with a foray into automated design, they can proceed to engage matters of aesthetic judgement and artistic technique.

For example, consider the following DESIGN program.

```
     ∇ PICTURE ← SIZE DESIGN COLORS
[1]    HOWMANY ← ⍴COLORS
[2]    PICTURE ← COLORS[?SIZE⍴HOWMANY]
     ∇
```

DESIGN uses some COLORS (symbols on the keyboard) and some SIZE (two dimensions of a matrix) to produce a PICTURE.

A simple program like DESIGN goes a long way with children. They seem never to tire of it, for it can produce quite a variety of designs:

```
      10  20  DESIGN ' O 0 ○ ∘ . '

○        OO    ∘ ○
OO   O  O○.∘ ∘   ∘○
0 0  O O ○O∘ 0O  .
 ∘    0∘∘O   0   ∘∘0∘
 ∘∘ ○O○0 ○      0O
      ∘    O .   0∘○
.0∘  ○  0 O○ .0 O
. 0     O○ O       ○
   ○0.       O     ○
.∘  . . .      .  ○.
```

```
      10  20  DESIGN  '⎕⎕* *⎕⎕'

*  *⎕⎕ *⎕⎕⎕⎕⎕*⎕⎕⎕**⎕
 ⎕⎕⎕⎕ ⎕  ⎕⎕⎕⎕* **⎕⎕*
*⎕⎕⎕⎕⎕ *⎕⎕⎕*⎕* ⎕⎕*⎕⎕
*⎕ ⎕⎕⎕⎕⎕*⎕ ⎕⎕⎕*⎕⎕
⎕⎕***⎕⎕⎕**⎕⎕ ⎕⎕*⎕⎕⎕⎕
***⎕**⎕⎕**⎕⎕* ⎕⎕*⎕⎕⎕
* *****⎕⎕* **⎕⎕*⎕⎕*⎕
*⎕*⎕⎕*⎕⎕*⎕*⎕*⎕  ⎕*⎕*
 *⎕ *⎕⎕⎕* ⎕⎕*⎕ *⎕⎕⎕⎕
⎕⎕⎕⎕⎕⎕⎕⎕⎕*⎕* ⎕⎕*⎕⎕⎕⎕
```

```
      10  20  DESIGN  '//Δ●∇\\'

Δ/Δ∇//Δ∇\∇\∇\∇\\●/\\
∇/●∇∇Δ∇/∇\/Δ\///Δ/\∇
\∇∇///∇●/\●\\/\\●\∇\
∇/∇●/\Δ//∇\//\//∇Δ∇∇
Δ∇\\●∇Δ\∇\/∇/Δ∇/\\\\
/●●//Δ∇●///\Δ/●/∇Δ●●
//Δ\∇/\∇/∇\/\●●Δ/∇\\
∇∇∇\\●\●∇/\Δ∇Δ●ΔΔΔΔ/
\/●\∇∇∇Δ\●∇\//\●\∇Δ\
∇●Δ\\\∇∇\Δ●∇\Δ\\\Δ\Δ
```

```
      10  20  DESIGN  ' ~ '

  ~     ~     ~  ~ ~
 ~~ ~~~         ~ ~
   ~~  ~~~  ~  ~ ~
   ~       ~ ~~ ~~
 ~     ~ ~
 ~ ~~~ ~~~  ~~    ~
~ ~~ ~ ~    ~  ~~
  ~ ~    ~ ~      ~
    ~  ~     ~
~    ~~ ~~~
```

```
      10  20  DESIGN   ' ¯-||-_ '

| -|¯-||_|-¯   ¯| --
_|-_   ¯  - -- _ ¯ ||
-_  |¯   _ -_- | - -
  |__ -  ||  |  |_ ¯|-
_|  | __  ||_||--| _-
-   _  _    - _  | |
|-_|||| |¯ -|_   -|
| |--_--|¯ ||   _--_
|-|  -  ¯|__|  -- -
-  - -¯||¯-_-|¯--¯--
```

While these "computer hieroglyphics" may have dubious aesthetic appeal, one can imagine -- instead of these typed symbols -- randomly generated swatches of color, perhaps displayed on a televsion-like screen.

Extensions of this approach to computer art include: (a) automating DESIGN, (b) weighting the selection of COLORS, (c) asking for human judgement (Do you like it or not?) in order to adjust weights on COLORS or other aesthetic factors, and (c) piecing together several computer-generated PICTUREs into a montage.

Another approach to computer art involves viewing programs which simulate artistic technique. For example, consider the program MONDRIAN below (named after the Dutch abstract painter).

```
     ∇ MONDRIAN
[1]    CANVAS ← 30 50 ⍴ ' '
[2]    DAB: COLOR ← 'o⎕*'[?3]
[3]    SIZE ← 3 5 ⌈ ?6 10
[4]    PICK: PLACE ← ?30 50 - SIZE
[5]    OVERLAP ← +/+/CANVAS[PLACE[1]+⍳SIZE[1];PLACE[2]+⍳SIZE[2]]≠' '
[6]    →PICK IF OVERLAP>2
[7]    CANVAS[PLACE[1]+⍳SIZE[1];PLACE[2]+⍳SIZE[2]] ← COLOR
[8]    →DAB IF (PERCENT ' ' ON CANVAS) > 67
[9]    CANVAS
     ∇
```

MONDRIAN begins with a blank canvas (arbitrarily set at 30 by 50). Then the program chooses a random COLOR, SIZE and PLACE to DAB.

OVERLAP measures the extent of overlap with DABs already on the CANVAS.

IF OVERLAP is greater than 2, then it will PICK another PLACE. (This is tantamount to finding relatively open space on the CANVAS).

IF, however, OVERLAP is not too large, the COLOR is put on the CANVAS at the PLACE and in the SIZE selected.

The program continues to DAB IF the PERCENT of blank spaces ON the CANVAS is greater than 67. In other words, as soon as it is 1/3 filled up, CANVAS is displayed.

Note: MONDRIAN uses two simple sub-programs (mostly for readability). They are PERCENT and ON:

```
     ∇ HUNDREDTHS ← PERCENT  N
[1]    HUNDREDTHS ← ⌊0.5 + 100×N
     ∇

     ∇ DENSITY ← SYMBOL  ON  PICTURE
[1]    DENSITY ← (+/+/SYMBOL=PICTURE) ÷ (×/⍴PICTURE)
     ∇
```

Now, MONDRIAN at work:

```
      MONDRIAN

                    **********
                    **********            ⎕⎕⎕⎕⎕
      oooooooooo    **********            ⎕⎕⎕⎕⎕
      oooooooooo    **********ooooo       ⎕⎕⎕⎕⎕
      oooooooooo              ooooo
 ⎕⎕⎕⎕⎕⎕                       ooooo
 ⎕⎕⎕⎕⎕⎕                    ⎕⎕⎕⎕⎕⎕⎕
 ⎕⎕⎕⎕⎕⎕                    ⎕⎕⎕⎕⎕⎕⎕
        ooooooo            ⎕⎕⎕⎕⎕⎕⎕     ⎕⎕⎕⎕⎕⎕⎕⎕
        ooooooo            ⎕⎕⎕⎕⎕⎕⎕     ⎕⎕⎕⎕⎕⎕⎕⎕
        ooooooo   ⎕⎕⎕⎕⎕    ⎕⎕⎕⎕⎕⎕⎕     ⎕⎕⎕⎕⎕⎕⎕⎕
                  ⎕⎕⎕⎕⎕    ⎕⎕⎕⎕⎕⎕⎕     ⎕⎕⎕⎕⎕⎕⎕⎕
        ⎕⎕⎕⎕⎕⎕⎕⎕⎕ ⎕⎕⎕⎕⎕       *****    ⎕⎕⎕⎕⎕⎕⎕⎕
*****   ⎕⎕⎕⎕⎕⎕⎕⎕⎕             *****    ⎕⎕⎕⎕⎕⎕⎕⎕
*****   ⎕⎕⎕⎕⎕⎕⎕⎕⎕             *****
*****                             ******
*****⎕⎕⎕⎕⎕                        ******
*****⎕⎕⎕⎕⎕                        ******
*****⎕⎕⎕⎕⎕         ********       ******
                   ********       ******
                   ********       **********
                   ********           ******
*****              ********           ******
*****  oooooooooo  ************       ******
*****  oooooooooo         *****       ******
       oooooooooo         *****       ******
```

Possible extensions of this kind of program include: (a) simulating and combining additional artistic techniques (those that can be operationalized), (b) computing abstract measures of difference between random "paintings" produced by the computer, (c) converging to minimal differences from a previously specified "ideal" painting, and (d) developing a model for aesthetic judgment -- perhaps one which "evolves."

Conclusion

These are but a few APL "glass box" programs designed to stimulate students to think about selected concepts. Each of the sample programs shown here can be used *as is* and, of course, can be extended in a myriad of directions. Other topics well-suited for this pedagogical approach include some drawn from linguistics, statistics, mathe-

Creative Chess

by Walter Koetke
Lexington High School, Mass.

The game of chess was introduced to Europe near the middle of the thirteenth century. As one might expect of a game with over 197,000 ways of making the first four moves with over 71,000 resulting positions, many people have always been fascinated with the game but so very few have mastered it. This article is not, however, about the game of chess but rather about a few of the recreations it has spawned.

According to W.W. Ball (1), one of the oldest known chess related recreations was proposed in the early 1500's. The problem consisted of a 3 by 3 board with two white knights (W) and two black knights (B) positioned as shown.

B		B
W		W

The object is to move the pieces so that the squares occupied by the white knights are occupied by the black knights and vice versa.

When one is learning to play chess, some type of point value for each piece is usually assumed. For example, I was taught to value a queen as 10, a castle as 5, and a bishop and knight each as 3. You may have noticed that different introductory texts are likely to suggest different point values. How can this be? How are these point values assigned? If the relative point values are valid, then why do the experts disagree?

Suppose we choose to evaluate a piece by computing its "checking power". We can define the checking power of a piece as the probability that the piece will have the king in check if the piece and the opposing king are placed at random on an empty board.

To compute the checking power of a castle, the castle is placed on an empty board leaving 63 empty squares. Of these 63 squares, 14 are controlled by the castle. A king placed at random on the board would, therefore, have a probability of 14/63=2/9 of being in check. Thus the checking power of a castle is 2/9.

Computing the checking power of a bishop is a bit more tedious. A bishop placed on any of the 28 squares in the outer ring of the board commands 7 of the remaining 63 squares. However, if the bishop is placed on any of the 20 squares in the ring of squares adjacent to the border ring, then it commands 9 of the remaining 63 squares. Similarly, if placed on one of the 12 squares of the next inner ring the bishop commands 11 of those remaining, and if placed on one of the 4 center squares the bishop commands 13 of those remaining. Thus the checking power of a bishop is:

$$\frac{28}{64} * \frac{7}{63} + \frac{20}{64} * \frac{9}{63} + \frac{12}{64} * \frac{11}{63} + \frac{4}{64} * \frac{13}{63} = \frac{5}{36}$$

Using similar logic, the checking power of a knight can be computed as 1/12 and the checking power of a queen as 13/36. Converting the computed checking power to integers yields:

Piece	*Checking Power*	*Relative Point Value*
Queen	13/36	13
Castle	2/9	8
Bishop	5/36	5
Knight	1/12	3

What do these relative point values suggest? Perhaps that my own assumed values of 10, 5, 3, and 3 aren't very good. Perhaps that our definition of "checking power" should be improved. Perhaps that the chess masters are supplying relative point values from their experience rather than from reproducible computation.

Try computing some relative point values on your own. Can you define "checking power" so you can produce the relative point values you've assumed? One alternate definition of checking power begins as ours did, but then excludes these squares from the pieces control from which the king could take the piece. For example, consider the computing power of a castle using this alternate definition. If the castle is placed in any one of the 4 corners, it controls 14 of 63 remaining squares. However, if the king was placed in either of the 2 squares adjacent to the castle it could take the castle. Thus the castle really only controls 12 of the 63 remaining squares. If the castle is placed in any of the other 24 outside squares it again appears to control 14 squares, but 3 of these, those adjacent to the rook, must be excluded because the king could take the rook from these positions. Similarly, if the

Computer Glass Box continued —

matics, engineering, ecology, and physical sciences.

The challenge to educators, then, is to identify such topics suitable for embodiment as glass box programs, to search out the kernel concepts to be taught, and to lead students to better understandings of those concepts using a programming language.

References

[1] Papert, S. "Teaching Children Thinking", M.I.T. LOGO Memo No. 2, Oct. 1971.
[2] Iverson, K. E. "APL in Exposition", IBM Tech. Report No. 320-3010, Jan. 1972.
[3] Berry, P. et. al. "APL and Insight: The Use of Programs to Represent Concepts in Teaching", IBM Tech. Report No. 320-3020, March 1973.

castle is placed in any of the 36 remaining squares it appears to control 14 squares 4 of which must be excluded. Thus the revised computation power of a castle is

$$\frac{4}{64} * \frac{12}{63} + \frac{24}{64} * \frac{11}{63} + \frac{36}{64} * \frac{10}{63} - \frac{1}{6}$$

and so forth. The remaining computations and further modifications of the definition of checking power are left to you.

An article discussing chess related recreations would be incomplete without considering some of the many "tour" problems, particularly since they offer interesting programming challenges as well. There are many different types of tour questions—the fewest number of moves to accomplish a task, covering all squares once and only once, the minimum distance traveled, . . . Here we'll consider just three different questions.

First, can you determine a path traveled by a king such that each square is occupied once and only once? No, it's not that easy. There's one additional catch. As the king moves, number each square consecutively starting with 1. When you've finished, the squares will each contain one of the integers 1 thru 64. What's the catch? The resulting board with numbered squares must be an 8X8 magic square. A solution will be published in a future issue.

As a second problem, try moving the castle from one corner of the board to the diagonally opposite corner, again passing through each square once and only once. This one isn't quite as easy as a quick reading suggests, but you should be able to solve it in a reasonable period of time.

Finally, consider the standard "Knight's Tour" problem that is considered in so many articles on recreational chess. A knight must cover the entire board occupying each square once and only once. Don't, however attack this one with a knight and a chessboard—try something different. Write a problem that will find a solution for you. Kemeny (2) offers a BASIC program that attempts to find a solution by making random moves. The small sample of runs he published doesn't include a complete solution, in fact not one of 25 runs exceeded 50 squares before no moves were possible. Perhaps surprisingly, however, in 15 of the 25 runs the tour did pass thru over half the squares before terminating. Actually, when a knight's tour is attempted using only random moves, long tours are quite common but a complete tour is very unlikely indeed. How then can a program help? By adding a little bit more to the move selection than simply the choosing of a random number. When selecting a move, add one additional criteria. Always move to a square from which the knight will command the fewest squares that have not been occupied. Fewest is not a misprint, even if it does contradict your intuition. If several squares fit this criteria equally well, then select one of them at random. This additional criteria is reasonably implemented in a program, and although it doesn't guarantee a successful tour, you are apt to be surprised by the results!

Bibliography

1) Ball, W.W. Rouse, *Mathematical Recreations and Essays,* Macmillan and Co., London 1940
2) Kemeny, J.G. and Kurtz, T.E., *BASK Programming* Second Edition, John Wiley and Sons, New York, 1971

Big Surprise From Small Computers in Chess Matches

Good things come in small packages—especially when computers compete in chess matches. In two matches here and abroad, small computers performed surprisingly well against giant competitors.

David actually conquered Goliath in an intramural chess match at Columbia University when a small Data General Supernova—about the size of an attache case—checkmated an IBM System 360/91—one of the largest in the world—in just 25 moves.

The Supernova, owned by Columbia's Department of Electrical Engineering and Computer Science, had a memory capacity of 32,768 bytes; the IBM system, part of the university's computer center, had a capacity of over 2 million bytes.

The Supernova's chess program was written by Professor Monroe Newborn of the Electrical Engineering and Computer Science Department and by student George Arnold. It is written in assembly language and uses a technique that determines the best move by searching between four and eight half-moves ahead, selectively analyzing about 1,000 terminal positions. A move is determined in about one minute—well within chess tournament rules.

The game lasted more than 90 minutes, but the Supernova gained a decided advantage on the sixth move: the System 360/91, playing white, blundered, and traded a knight for a pawn. One of the program authors noted that the computer saw the correct move (bishop takes bishop) but didn't realize that exchanging bishops would save the knight. Once the big computer decided it could not save the knight, it decided to pick up a pawn.

"Having exclusive use of the smaller computer, along with running the program on-line, helps offset the greater speed and capacity of the larger machines," Professor Newborn said.

In another surprising match, a Computer Automation Naked Mini computer using only 16,000 words of 16-bit memory came in 12th in stiff competition in the World Computer Chess Championships held in Stockholm, Sweden.

"I'm no expert at chess; in fact, I'm just an average amateur, but I love to play with computers. Even so, I was surprised, indeed," said Bob Prisen, Interscan Data Systems, Ltd., United Kingdom, who programmed the Naked Mini. He spent approximately 300 hours in an eight-month period and used BASIC assembler language.

The first-place winner, by contrast, was a large-scale English-built ICL 4/70 computer entered by the Moscow Institute of Control Science. The machine and its programming had been prepared by a team of 10 fulltime staffers for more than two years. The computer used a program called KAISSA.

During the Swedish match, the Naked Mini stayed in England. Communications between Prisen and the computer were established using international telephone lines and an acoustic coupler with a Teletype and an on-line visual display unit.

Prisen said he hopes that a European or British Chess Computer Championship can be arranged in the future. "If it occurs, I'm confident the Naked Mini computer will greatly improve its position in the Chess Computer League with a bit more core memory and programming effort," he said.

About Computing

by Geoffrey Chase, OSB
Portsmouth Abbey School, RI

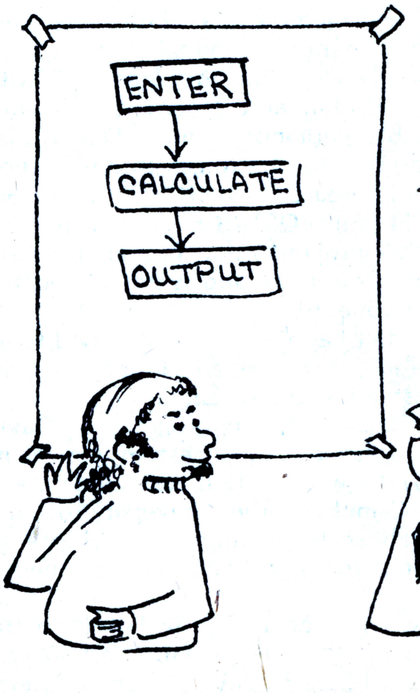
There's a famous "theorem" that runs something like this:

1. Any non-trivial program has at least one "bug".
2. (Corollary) Any program that works is trivial.

Computing—i.e., giving a set of instructions ("program") to a machine telling it what to do and what paths to take *if* this rather than that happens, and when to stop—is different from what most people seem to expect. To begin with, it doesn't necessarily have anything much to do with numbers, arithmetic, or algebra, though students of these subjects often use computers and usually (by no means always) find programming easier than do people who haven't much math background.

Another odd feature is that it is not quite so much a matter of "right" or "wrong" answers as one might think. Almost no one gets his program right the first time he runs it, unless it's copied off a book or does nothing you couldn't do better without the machine. And of those who do get the "right answer"—i.e., a program that finally works—some will write programs that are unnecessarily slow, use up too much of the machine's memory, and stretch an accumulation of small errors into large ones [computers don't make mistakes but they often generate error—figure that one out!]. Someone else may write a program that is lean, taut, elegant, fast and accurate. Both have come up with the "right answer", but the second program is obviously "righter" than the first. Often it is also simpler, or simpler-*looking*, than the inferior program.

The hard part of computing is keeping your head clear. The easy part is what looks hard, programming in some sort of language.



I'd expected a little
more documentation.

Some more "theorems", stated without proof:

3. Any fool can write a program no one else can understand. It takes intelligence to write one that is clear and seems easy.
4. Our #3. fool will find, six weeks later, that even *he* can't figure it out.

Write programs so someone else can read them. Use a few comments; make the order of the program adhere as near as can be to the order in which one *thinks* about the problem being solved.

5. Computers are *dynamic*—they "move" things. Many familiar expressions in math and elsewhere are *static,* reflecting an unchanging truth but not giving much hint of how to find or figure. So your job in most instances is to find some sort of repeated *process* (they call computers "processors" sometimes) which will get you what you want in a finite number of "moves" or steps.

Such a process is called an "algorithm". You often don't know, by the way, *how many* moves or steps will be required; merely that there will be only a finite number of them.

It's not at all rare to write an "infinite program" by mistake. This can hang up the processor and the programs of other users, who will be tempted perhaps to hanging of another sort . . .

Example: $Y = 5 \cdot x^3 + 4 \cdot x^2 + 3 \cdot x + 2$ and $X =$ (let's say) 1.2; how do we find the value of "Y"?

"Static" method: multiply X times X times X times 5 and then . . . all the way down the line. A lot of work.

"Dynamic method": let $Y = 5$. Now, for "K" equal to 2, then 1, then 0, do the following:

a. Multiply Y times X (which is 1.2) and add the coefficient of X-to-the-K-power. When K reaches 0, this is the so-called "constant term", +2 in our problem. Let Y = this new value and *throw away* the old value of Y. In computer talk,
$Y = Y{*}X + C(K)$, where "*" means "times" and C(3), C(2), C(1), C(0) are the coefficients (from left to right) of our equation. Note that C(K) means "C subscript K"—not "times K"—and that the = sign really means "is replaced by" rather than "is identical to".
b. Your last Y is the answer. Believe it or not, this is by far the easier way to do our problem.

This repeated application of a simple idea is the key to programming.

6. A woman's work is never done, and neither is the programmer's.

You will nearly always find yourself rewriting programs. It might work, *but* you would like another feature added; or, it blows up if the user does something stupid so let's check for stupidity on his part; or, it can be combined with two or three other short programs to make the Pan-galactic Interplanetary Super Solver that cures all ailments.

It's like fine furniture: sanding, resanding, finishing, refinishing, until you really like the looks of it.

7. When in doubt, guess. You don't know what will happen if . . . ? Well, try it. Smart guess-work

David vs. 12 Goliaths

by Monty Newborn

It looked like a sure thing; David would quickly polish off his opponents and we would be able to have an early dinner—most likely before midnight! The slugfests between programs to determine a national champion usually begin at 7 p.m. at the ACM Annual Conferences and continue until almost dawn—certainly well past midnight—and for those participating, this means a late gourmet meal at a nearby twenty-four hour diner over which the evening's activities are rehashed. But tonight, rather than battling each other, the twelve programs were scheduled to take on David Levy, British International Chess Master, in the first simultaneous chess exhibition in history in which a Master could not count on his opponents cringing in respect. Dinner was not too far off.

The scene was a second floor conference room in the Radisson Hotel in downtown Minneapolis; the date was October 19, 1975. The audience of several hundred included both chess experts and computer professionals. On stage were the authors along with teletypes and telephones connecting to remote computers. My role was that of organizer, along with Ben Mittman of Northwestern University, and participant. Data General had been good enough to provide me with a Nova 2 for the event and it served as the only "live" and visible entrant, its lights twinkling as moves for my program OSTRICH were calculated. David has been serving as director of the ACM Tournaments since 1971 and is likely to come into several thousand dollars in 1978 when the computer community fails to produce a program that can defeat him in a match. He accepted a wager in 1968 to this effect from a distinguished group of computer scientists. This simultaneous exhibition would certainly set straight the supporters of the metal monsters!

And so the evening began. It became clear that David intended to take the computers "out of book" as soon as possible and, in general, play somewhat closed positional games. The computers played slowly, taking about 3 minutes per move, while David bounced from one board to another, only seeming to be concerned over his game versus CHESS 4.4 running on a CDC CYBER 175. OSTRICH was holding ground but definitely having the worse of it. One by one David's opponents met defeat, marked visibly by the disconnected telephone and the posting of the results on large display boards in the room. But the games were lasting longer than I expected. David must be playing very safe was my guess; he doesn't want to lose any games or draw any either. The pressure against OSTRICH continued to grow with David building up small gains. But it was now nearing 10 p.m. and there were still about six programs alive! I became completely immersed in my game at this point, losing track of what was happening in the other games. David seemed to be only a few moves from crushing OSTRICH and I was glad to see OSTRICH had made a run of it at least. But then, much to my surprise, David made a weak 25th move giving OSTRICH some chance for equalizing the game and even a chance to gain the lead. OSTRICH saw David's error and made the correct reply, and David was in trouble—but not enough. David, playing at a slight material disadvantage, gradually recaptured the lead and defeated OSTRICH on move 50. Thus, at about 11 p.m. OSTRICH joined the ranks of about 8 other programs that had gone down to defeat. I could now relax and watch David finish off his other opponents.

But it didn't happen that way. David found himself behind in two games and fought to survive until well past midnight when his opponents agreed to draws. CHESS 4.4, the program of David Slate and Larry Atkin of Northwestern University, and TREEFROG, the work of Ron Hansen, Russell Crook, and Gary Calnek of the University of Waterloo, both were ahead but unable to develop a strategic plan leading to a resignation by their mortal opponent.

So once again at 1 a.m. we went to the local diner to rehash the day's events and speculate on how numbered David's days are. On a head-to-head match, ten years may still be safe, but David—beware!

"About Computing" continued —

is much more important in (even) pure mathematics, to say nothing of more practical things, than people realize. Every theorem was once merely a hunch. It is important to find its proof, true; but it is almost more important to find something to prove!

8. Aesthetics (elegance, style) count for a lot, and are a lot more practical than "gimmicks" in the long run.

9. Computer books and write-ups (this one included) are hard to read at first, and full of words that nobody bothers to explain.

Suggestion: If you have trouble on this score, plow ahead and keep on reading, regardless of whether you know what the author is saying. A later page may very well explain an earlier one. Then go back to what bothered you; if it still bothers you, make a note of it and lay the book down.

Another day it may come clear; or somebody may be able to answer your questions.

The writer first met computers when he was 40. The result: near-despair until the above technique was used; then things began to come clear. It's like learning a foreign language; speak and hear as much of it as you can, even if half is obscure.

10. The ingenious chaps who invent computer languages (like BASIC, which will probably be what you start with) put in error "traps" to catch certain obvious blunders. But in general, alas,

a. The machine can and will do what you ask it to do, no matter how stupid or wise your orders may be. It doesn't know the difference.

b. The machine can't possibly tell you *what* you should be asking it to do. Of fact and fiction, value and meaning, it knows nothing. "GIGO" (garbage in, garbage out) is an old IBM motto!

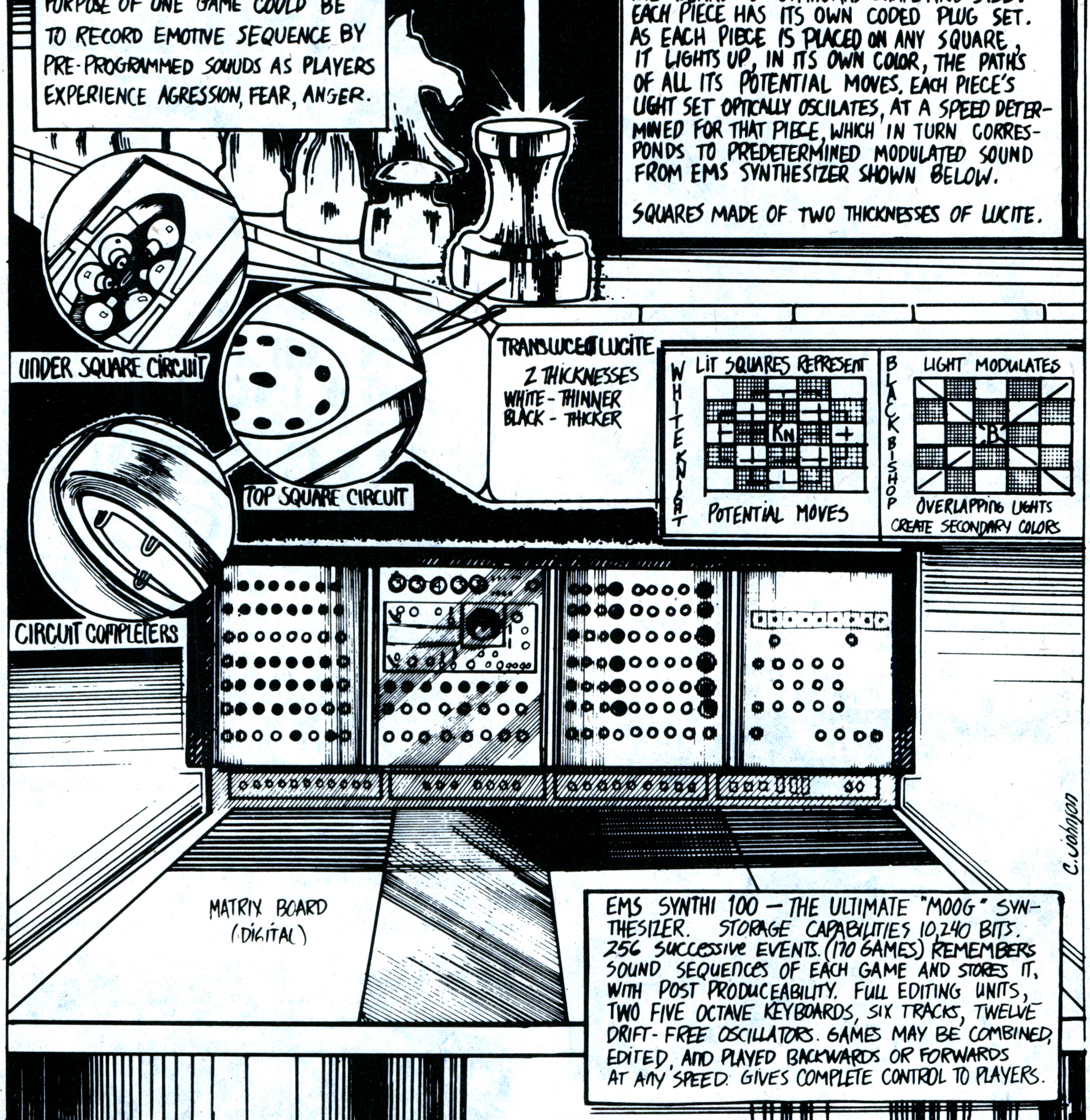
Good luck on your adventure!

CONCEPTUAL CHESS

THE CONCEPT: EACH GAME PLAYED WITH THIS SET-UP EVOLVES INTO A DIFFERENT DISPLAY OF IMAGE OCCURENCES THRU MODULATED SOUND AND COLORED LIGHT. EXAMPLE: PURPOSE OF ONE GAME COULD BE TO RECORD EMOTIVE SEQUENCE BY PRE-PROGRAMMED SOUNDS AS PLAYERS EXPERIENCE AGRESSION, FEAR, ANGER.

THE BOARD IS STANDARD SHAPE AND SIZE. EACH PIECE HAS ITS OWN CODED PLUG SET. AS EACH PIECE IS PLACED ON ANY SQUARE, IT LIGHTS UP, IN ITS OWN COLOR, THE PATHS OF ALL ITS POTENTIAL MOVES. EACH PIECE'S LIGHT SET OPTICALLY OSCILATES, AT A SPEED DETERMINED FOR THAT PIECE, WHICH IN TURN CORRESPONDS TO PREDETERMINED MODULATED SOUND FROM EMS SYNTHESIZER SHOWN BELOW.

SQUARES MADE OF TWO THICKNESSES OF LUCITE.


UNDER SQUARE CIRCUIT
TOP SQUARE CIRCUIT
CIRCUIT COMPLETERS
TRANSLUCENT LUCITE
2 THICKNESSES
WHITE - THINNER
BLACK - THICKER
WHITE KNIGHT
LIT SQUARES REPRESENT
Kn
POTENTIAL MOVES
BLACK BISHOP
LIGHT MODULATES
B
OVERLAPPING LIGHTS CREATE SECONDARY COLORS
MATRIX BOARD (DIGITAL)
C. Johnson

EMS SYNTHI 100 — THE ULTIMATE "MOOG" SYNTHESIZER. STORAGE CAPABILITIES 10,240 BITS. 256 SUCCESSIVE EVENTS. (170 GAMES) REMEMBERS SOUND SEQUENCES OF EACH GAME AND STORES IT, WITH POST PRODUCEABILITY. FULL EDITING UNITS, TWO FIVE OCTAVE KEYBOARDS, SIX TRACKS, TWELVE DRIFT-FREE OSCILLATORS. GAMES MAY BE COMBINED, EDITED, AND PLAYED BACKWARDS OR FORWARDS AT ANY SPEED. GIVES COMPLETE CONTROL TO PLAYERS.

THE INCREDIBLE VARIETY OF TONE PRODUCEABILITY ENABLES ONE TO INSERT IDENTIFIABLE CHORDANTS INTO SEEMINGLY RANDOM SEQUENCES. I.E. WHITE BISHOP GIVEN SOUND OX), TAKES BLACK PAWN — (GIVEN SOUND OY), TO PRODUCE SOUND OXY (DISCHORDANT) TO DENOTE "CHAGRIN."

AT CHECKMATE, COMPUTER IS PROGRAMMED TO DETONATE STICK OF DYNAMITE UNDER LOSER'S CHAIR. (KA-BOOM!)

$19.95 RET.

A Brief Guide to the Theory of Relativity

(in easy-to-understand language)

by Peter Payack

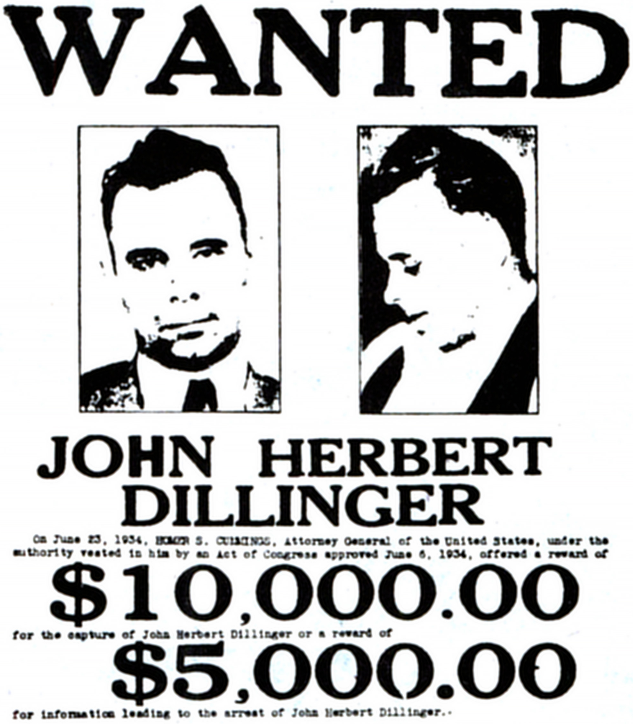
When the theory of relativity was first propagated by Albert Einstein in 1905, it was said that only twelve men could understand it. Unfortunately that was 70 years ago, and they are probably all dead now. So we cannot look to them for any help, but must rely on our own resources.

A short time ago it came to me in a dream or vision or television commercial (I can't remember which) to use the metaphor of a train when describing the mechanics of relativity. The Long Island Railroad is the perfect example.

One of the paradoxes of relativity is that of time. Picture yourself on a Long Island Railroad commuter train at rush hour. There is a clock on the station wall, and further suppose that it works. Remember a vivid imagination is important. The clock on the wall says 5:00 when the train pulls out at the speed of light. After one second the train has traveled 186,000 miles, or past Hempstead anyway. To the people on the train it will seem as if one second has passed, and yet if they look back at the clock on the station wall, it still says 5:00. Time on the train has come to a stop! At first glance it may seem that we are dealing with a purely logical contradiction. But certainly this is not so for anyone who has ever ridden on the Long Island Railroad, where time always seems stopped.

This conclusively proves that there is no universal time, and even though it might seem so to some New Yorkers, the universe does not run on Eastern Standard Time.

Another paradox we are confronted with is the length contradiction. Some say that John Dillinger had the longest one at 18 inches, and that the proof is in the Smithsonian.



WANTED

JOHN HERBERT DILLINGER

On June 23, 1934, HOMER S. CUMMINGS, Attorney General of the United States, under the authority vested in him by an Act of Congress approved June 6, 1934, offered a reward of

$10,000.00

for the capture of John Herbert Dillinger or a reward of

$5,000.00

for information leading to the arrest of John Herbert Dillinger.

Personally, I doubt this claim. Nevertheless, when we continue our train ride at the speed of light, hopefully avoiding all delays, we would find some amazing changes in appearances as we pass through stations along the way. People in the stations will think that the train has become shorter while to us in the train, it will appear that the platforms are thinner and taller. This is not unusual right after cocktails. The only person things won't seem distorted to is John Dillinger, and that's because he's dead. To a dead person, length, width and even relativity don't matter much.

In a straightforward manner, then, the theory of relativity provides us with a most dizzying picture of the world about us. Unfortunately for us on this train, technology has not increased as rapidly as theory, and no brakes were provided for stopping a train at 186,000 miles per second.

Puzzles

15 BIG pages containing 115 PUZZLES for computers, calculators, and humans!

by David H. Ahl

How does one solve a puzzle or problem? There are direct and indirect approaches. Frequently, the "scientific method" is advocated as the generalized approach to solving all kinds of problems. Stated briefly, the scientific method consists of the following steps:

1. State the problem. Break it down into manageable pieces, if necessary.
2. Collect facts and data.
3. Using the data, try a solution. Does it meet the objectives, i.e., does it solve the problem? If so, is it the best solution? If not, go back to Step 2.

But how do we get the solution from the data? Several possibilities exist:

1. *Deduction.* Reaching a conclusion from something already known.
2. *Inference.* Reaching a conclusion from facts and evidence.
3. *Trial and Error.* Keeping at it, avoiding past mistakes until you get it right.
4. *Experimentation.* Trying something new and observing the results to achieve a goal.
5. *Intuition.* Direct perception of the truth that bypasses analysis.

In the puzzles and problems which follow, you'll probably have to use all five of the above methods at one time or another. Sometimes a calculator or computer will come in handy—but it's up to you to decide when you need outside aids. And although you can write a computer program to solve some of the problems, you have to decide whether the programming effort is "worth it," i.e., would it take less total time to solve it by pencil, paper, and human brain than with the assist of a machine.

I assembled this collection of puzzles and problems from about 20 sources which are cited at the end, plus a pile of original puzzles which I've been writing and accumulating in a dog-eared folder for years. If you like this kind of stuff, let me know, and we'll run more.

Mathematical Puzzles and Pastimes, Philip Haber (Ed.); The Peter Pauper Press, Mt. Vernon, NY; 1957. $1.95.

Pencil Puzzles & Word Games, 5th Edition, Dell Publishing, New York; 1975. 35c.

The Math Entertainer, Philip Heafford, Emerson Books, Buchanan, New York; 1959. $5.95.

150 Puzzles in Crypt Arithmetic, Maxey Brooke, Dover Publications, New York; 1963. $1.25. (Available thru *Creative Computing* Library.)

101 Puzzles in Thought and Logic, C. R. Wylie, Jr., Dover Publications, New York; 1957. $1.35.

Invitation to Mathematics, W. H. Glenn and D. A. Johnson, Dover Publications, New York; 1960. $3.50.

Pillow Problems and a Tangled Tale, Lewis Carroll, Dover Publications, New York; 1958. $1.50.

Fun With Mathematics, Jerome S. Meyer, World Publishing Co., Cleveland; 1952.

Mathematical Puzzles for Beginners & Enthusiasts, Geoffrey Mott-Smith, Dover Publications, New York; 1954. $1.75.

Computer Programming Problems, NCC Publications, U.K.; 1975. $0.75.

Gamesmag (Vol. 1, No. 1), Center for Open Learning, Box 9434, Berkeley, CA; 1975. $0.75.

Games & Puzzles, 11 Tottenham Court Road, London W1A 4XF, England. One-year subscription to USA $11.40. (One of the very best games magazines!)

The Mathematical Puzzles of Sam Lloyd, Vol. II, Martin Gardner (Ed.), Dover Publications, New York.

The Floodgate Scandal and Other Puzzles, Ivan Morris. (Preprint of a forthcoming book.)

Computer Problem Solving, R. P. Watkins, John Wiley (Australasia); 1975.

15 pages of puzzles ☛

SPEED TRAP

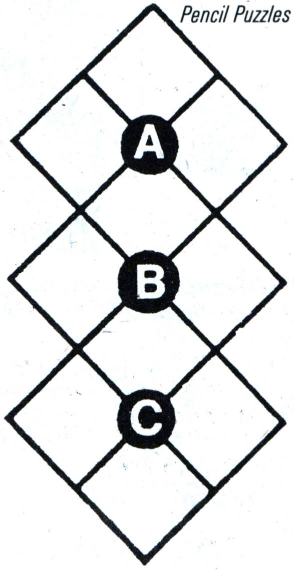
One day as a motorcycle cop was watching the street for speeders, he spied one coming down at breakneck speed about ¼ mile away. If the cycle covered 55 miles in an hour and caught up to the car in 300 seconds, how fast was the car traveling?

Mathematical Puzzles and Pastimes

NUMBER DIAMONDS

Can you distribute the numbers 1 through 10 (using each number only once) in the blank spaces in such a way that the numbers surrounding each letter—A, B, or C—will total 20? Several different combinations are possible.

Pencil Puzzles & Word Games



HOW LONG, MONET?

A water lily doubles itself in size each day. From the time its first leaf appeared to the time when the surface of the pond was completely covered took forty days. How long did it take for the pond to be half covered?

The Math Entertainer

GROOVY

The diameter of a long-playing record is 12 inches. The unused center has a diameter of 4 inches and there is a smooth outer edge 1 inch wide around the recording. If there are 91 grooves to the inch, how far does the needle move during the actual playing of the recording?

The Math Entertainer

ELECTRONIC LULLABY

Our small neighbor was given an electronics set for Christmas and we have had no peace since. His latest model is an electronic organ. Unfortunately, it only plays three notes, a high note, ping; a middle note, mmmmmm; and a low note, boing. He has wired these up so that the same note repeated; for example, ping ping is immediately followed by an mmmmmm. A note followed by a lower note is followed by ping, and a note preceding a higher note leads to a boing. Really quite impressive for a twelve-year-old, but it's getting on our nerves. Can you explain why?

Games & Puzzles

"ABRACADABRA"

An old numerologist from the Left Bank reports a secret Grecian formula for finding prime numbers (numbers divisible by nothing but 1 and themselves, like 7, 17, 19): $W^2 - W + 41 = P$.

For every whole number, W, P is prime. Correct?

WIRED UP

A telegraph wire is laid around the surface of the moon at its equator. A second telegraph wire is laid around the surface of the planet Mars at its equator. It is now required to raise each of these wires six feet above the equators. Clearly, in each case, more wire is required to do this, but how much more for each of the two bodies?

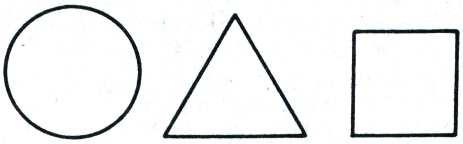
You may assume that Mars and the moon are both perfect spheres, having respective diameters of 4140 miles and 2160 miles. You may also assume that when the wire is raised six feet off the surfaces concerned, it will not sag between whatever supporting poles there are.

Games & Puzzles

SQUARE ROTATION

This circle, square and triangle each have the same length perimeter. If the circle is rotated without slipping around the other two, in which case will it rotate through the largest number of degrees?

Games & Puzzles



FLOWER CHILDREN

If five girls pack five boxes of flowers in five minutes, how many girls are required to pack fifty boxes in fifty minutes?

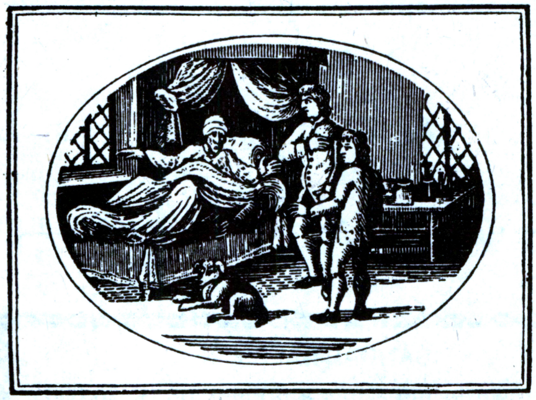
The Math Entertainer

IMPOSSIBLE DIVISION

Aunt Jenny had 3 greedy nephews—Philip, Sam and George, who eagerly looked forward to the day when she would die and leave her money to them. But Aunt Jenny decided to play them a little trick. She called in her lawyer one day and made out her will as follows:

The total estate amounting to $1,717 is to be shared by the nephews as follows:

Philip is to receive ½, Sam ⅓ and George 1/9, with the proviso that each is to receive an amount in even dollars only, according to his share. Each nephew is to have 24 hours counting from the hour of death, to calculate the exact amount in dollars he is to receive. If in the event any calculated share amounts to dollars and cents, or if no exact amount in dollars is arrived at at the expiration period of 24 hours, the whole sum becomes forfeited and is to be bequeathed to a worthy charity designated under Paragraph 7 of said will.



The very next day, Aunt Jenny passed away and her lawyer called in her three nephews to hear him read the will. At the end of the reading, they started to calculate their shares, but to their consternation found that no matter how they figured their shares they could not make them come out in an even amount in dollars.

At the end of 23 hours and 50 minutes, they were desperate, but in the last 10 minutes all went well. How did they solve their problem?

Mathematical Puzzles and Pastimes

ALL NINES

What six digit number added to itself five times will give a total each time having the same digits as that number but differently arranged, and after the sixth addition will give a total of all nines?

FREDDY THE FROG

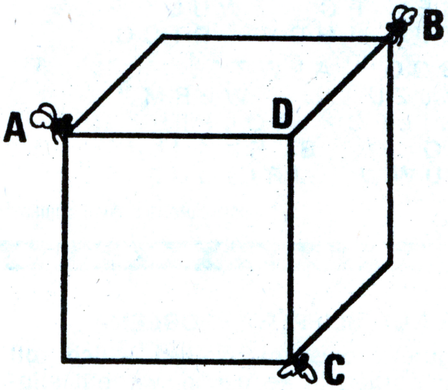
Freddy the Frog falls into a 99-foot well with a great splash, and at once starts climbing to the top. He goes up three feet every day and falls back two each night. Freddy continues in this indomitable fashion until he has reached the very top of the well. How long has it taken him?

Floodgate Scandal

FLIES ON CUBE

Three flies start at three corners of a cube, at A, B and C. The starting-gun fires and each starts crawling towards one of the others, all moving at identical speeds. A moves towards B, B to C and C to A. Each moves in the direction which would take it to its target in the shortest possible time *if* that target were stationary. Presumably they will end up arriving simultaneously at D, but in getting there will cross the edges of the cube, and if so, how many times?

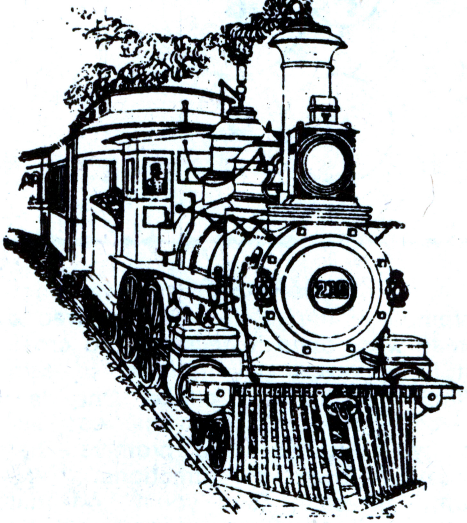
Games & Puzzles



JERSEY CENTRAL FLYER

From Philadelphia to Atlantic City is 60 miles. Two trains leave at 10:00 A.M., one train from Philadelphia at 40 miles an hour and the other from Atlantic City at 50 miles an hour. When they meet, are they nearer to Philadelphia or to Atlantic City?

The Math Entertainer



A CURIOUS PROPERTY

The number 142857 has many remarkable features. Here is one of the least known: $142857^2 = 20408122449$ and $142857 = 20408 + 122449$! There are four numbers of three figures with the same property. Square the number, add the number formed by the last three digits to the remaining number and the original number appears. One is the trivial 001; slightly less trivial is 999. And your puzzle is to find the other two. It may encourage you to know that their sum is exactly 1000.

Games & Puzzles

CRYPTOGRAMS

In these cryptograms, another letter of the alphabet has been substituted for the right letter. By noticing the frequency of certain letters (e, a, o, and n are usually the most often-used letters in English), and by looking for repeated patterns of letters in the words, you should be able to break each code. Each cryptogram is in a different code.

1. CTW EYDW GXH
FC PW PCDSTV
SE FC FWUU
XUU HCY RTCG.

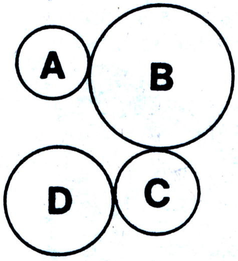
2. BOC ALWZG
KJZU J WLRM
GILPC GOLPI:
IOUC BJRI IL
IUWW JRLIOUP.

Pencil Puzzles & Word Games

A NON-SLIPPERY PROBLEM

The figure shows four rollers which roll against each other, in sequence, without slipping. The diameters of A and D are 4 inches and 6 inches respectively. B is twice the diameter of C and the total diameters of B and C together are equal to four times the diameter of A. If A is rotating at 3 revolutions per second, how fast will D rotate?

Games & Puzzles



CHARMER

In a remote country village there lived a poor woman who found it necessary to cross a certain bridge every day in order to earn her livelihood in the next village. One day, as she was about to cross, she was approached by a stranger who promised her riches if she followed his directions. "Take this charm," he said, "and you'll find that each time you cross, your money will be doubled. At the end of each day I'll be waiting for you to pay me ½ the money you then possess. But the charm will bring you this luck for only 3 days. Then you must return it to me or ill-luck will befall you."

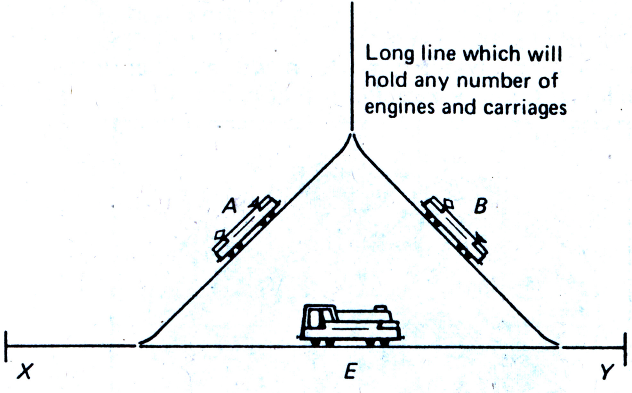
After 3 days, the stranger had collected 14 times as much money as the woman had had originally, while she came out $7.00 richer.

Could you find out how much the woman had originally and the amount the stranger was able to collect?

Mathematical Puzzles and Pastimes

SWITCH THE CARS

There are two railway carriages and an engine on the network of lines and sidings as shown below. As with all railway line intersections, points are used. Trains cannot go round corners, so they must move into a siding and then come out of it into the new track in the same order. The siding labelled *X* can hold either two carriages or a carriage and an engine, and nothing more. The siding labelled *Y* can only hold a carriage (an engine will not fit on it). Devise a method for shifting the engine and carriages around so that the carriages labelled *A* and *B* are swapped over and the engine *E* is returned to its original position. The engine only has a motor, and the carriages and engine can be linked in any reasonable way.


Long line which will hold any number of engines and carriages
A
B
X
E
Y

SANDAL SALES

A town in India has a population of 20,000 people. 5 percent of them are one-legged, and half the others go barefoot. How many sandals are worn in the town?

The Math Entertainer

M.C.P.

A boy is chosen president and a girl vice-president of the senior class of a school. In how many ways is this possible if the class has twelve boys and ten girls?

The Math Entertainer

SLUM HOUSING

A landlord owns a multiple dwelling housing project consisting of 2-room, 3-room, 4-room and 5-room apartments, renting for $60.00, $90.00, $110.00 and $120.00 respectively, per month. Altogether he has 100 apartments for which he receives $10,500 monthly from his tenants.

Find the number of apartments he can rent of each type.

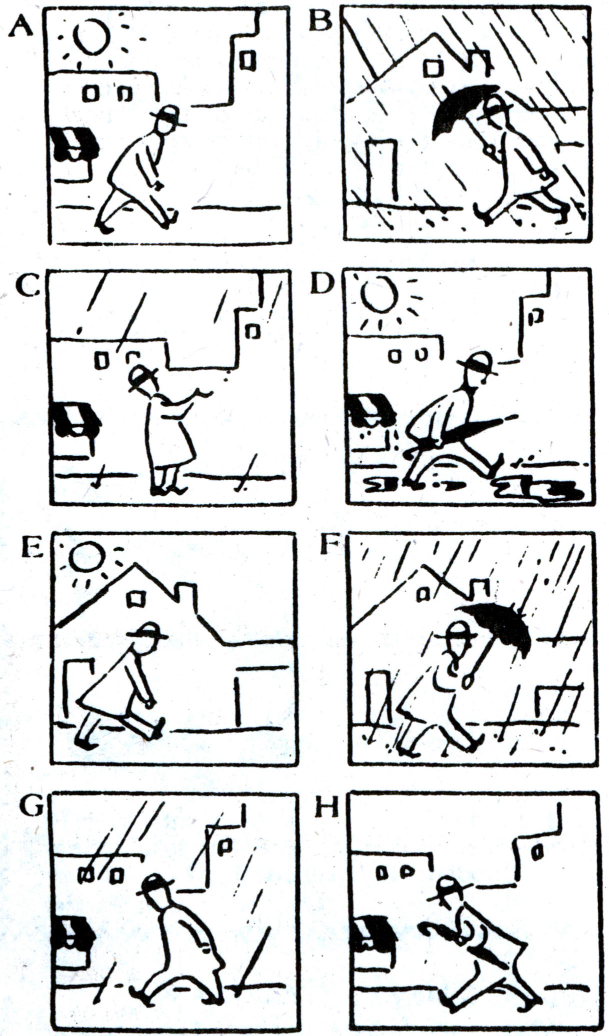
Mathematical Puzzles and Pastimes

SEQUENCE TEST

Study the series of pictures below. Then, starting with picture E, see if you can arrange them in the proper order so that they will present a logical sequence of events.

If you complete this test correctly, you rate very high in combining your power of observation with your power of deduction.

Pencil Puzzles & Word Games



ARTFUL ARITHMETIC

Johnny is not very keen on mathematics so he jumped at what seemed to him to be a quick way of discovering which of two fractions is the larger. Asked to find, for example, the larger of 2/5 and 3/7 he simply replaced them by 2/3 (2/5-2) and 3/4 (3/7-3) respectively, which he immediately replaced by 2/1 and 3/1 concluding triumphantly that the first, 2/5 is the smaller.

The teacher's problem, and yours, is of course to decide whether Johnny's method is valid, or whether it is nonsense, and his correct answer to this particular problem only a lucky fluke.

Games & Puzzles

HMMMM

$8^m = 32$. Find m.

SEVEN PAIRS

If two 1's, two 2's, and two 3's are arranged thus:

2 3 1 2 1 3

then the two 1's enclose 1 other digit, the two 2's enclose 2 other digits, and the two 3's enclose 3 other digits. Can you find a similar arrangement using the seven pairs 1,1; 2,2; . . . 7,7? Counting reflections, there are 52 different solutions, so it shouldn't be too hard finding one!

Games & Puzzles

MONEY, MONEY

One day Phil found to his dismay that he was short in funds by a certain amount. So he wired his sister Amy, as follows:

```
$  WI.RE
   MO.RE
$MON.EY
```

How much should Amy send Phil?

Mathematical Puzzles and Pastimes

THE BLETCHLEY LAWN TENNIS CLUB

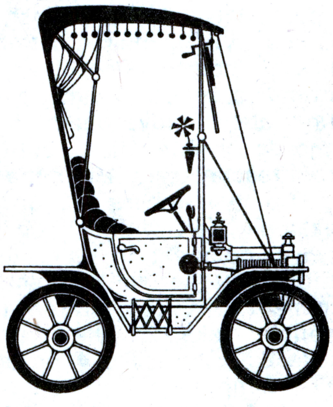
A total of 111 players belong to the Bletchley Lawn Tennis Club. What is the smallest number of games (all singles) necessary to determine who is the best player in the club? (Time limit: 30 seconds.)

Floodgate Scandal

A PRIME DEAL

Buying a new car is no joke, but my dealer was very understanding.

```
WE'LL
HE LP
WI TH
    A
-----
DE AL
```



he told me, 'And it's a prime deal, the best deal you could possible get.' What was the deal?

Games & Puzzles

PROBLEM FOR MA BELL

If the first three letters of a telephone number indicate the name of the exchange, how many such arrangements of three letters is it possible to devise from the twenty-six letters of the alphabet?

(*Afterthought: Without looking at a telephone, what two letters are not used on an actual dial?*)

The Math Entertainer

THE 100-BLOCK GANG

Bob and four other youngsters who live in the 100 block of Center Street have formed a group known as "The 100-Block Gang." Center Street runs east-west; numbers start at the west end of the block, with odd numbers on the south side of the street. There are five houses on each side of the block, so the numbers run from 101 to 110. From the following clues about the five and their families, you should be able to decide the full name and address of each member of the group.

1. The Baileys and the Prices live directly opposite each other in the center of the block.
2. John lives directly between Ethel and the Greens.
3. Vera lives directly across the street from Martha, and next to the Baileys.
4. The Berrys live immediately east of the Prices; there are no young people directly across from the Berrys.
5. The Golds live on the south side of the street, and John on the north side.

For this problem; we have presented a diagram of the 100 Block. We found it to be more of a solving aid than a regular chart would be.

Pencil Puzzles & Word Games

	102	104	106	108	110	
W						E
	101	103	105	107	109	

DOUBLE OR TAKE

This is a game for two players for which you need at the very most, pencil and paper. One player names a number. His opponent has a choice between doubling it, or taking from it a perfect square or a perfect cube. The first player plays again in the same manner and so on alternately, until one player on his turn reaches zero. He is the winner.

Some numbers are obviously losing. For example, left with 2, you have a choice between doubling to 4, or taking 1, leaving 1. Either way he will subtract a square and win immediately. Similarly 5 is a loss. If you double, take 1, take 4, your opponent will take 8 ($=2^3$), take 4 or take 1 respectively. Now suppose you are left with 21. Should you win or lose? If you are left with 17?

Games & Puzzles

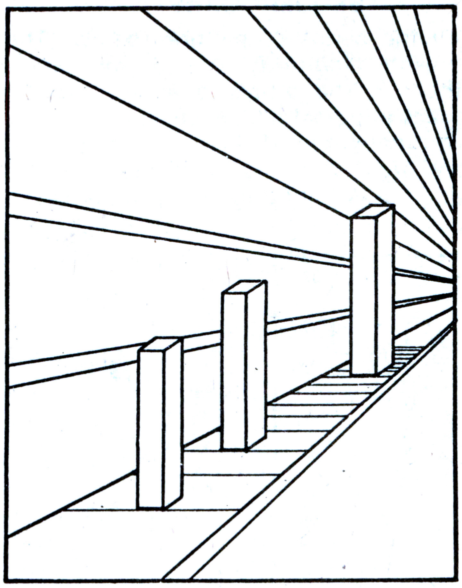
NUMBER COMBINATIONS

(A) Use three sixes to make 7.
(B) Use three fours to make 11.
(C) Use three threes to make 24.
(D) Use three fives to make 5.
(E) Use three fives to make 10.

Floodgate Scandal

NEXT?

What is the next number in this series: 92, 74, 46, 22, 18, . . .



The large oblong in rear is the same size as the others.

CATS & RATS

If 6 felines can devour 6 rodents in 1/10 of an hour, how many would it take to devour 100 rodents in 6,000 seconds?

Mathematical Puzzles and Pastimes

CHANGE FOR A DOLLAR

What is the largest amount of money you can have and still be unable to give change for a dollar? (Assume all your U.S. currency is in coins.)

E·PLURIBUS·UNUM
LIBERTY
1889

ZILCH—MILCH—PILCH

If 1 zilch is equal to 13 milches, and 1 milch is equal to 23 pilches, would you accept 8,000 pilches for 26 zilches?

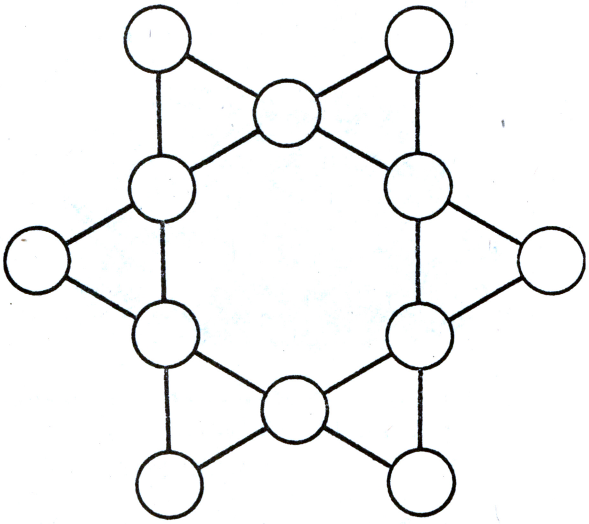
AND COMPUTER SCIENCE?

Among one hundred applicants for a certain technical position it was discovered that ten had never taken a course in chemistry or in physics. Seventy-five had taken at least one course in chemistry. Eighty-three had taken at least one course in physics.

How many of the applicants had had some work in both chemistry and physics?

101 Puzzles in Thought and Logic

MAGIC HEXAGON



Can you insert the numbers 1 to 12 in the diagram above so that the sum of the four numbers along each line totals 26? If you manage that successfully, now try it with the added restriction that the six points of the hexagon must add to 26 also!

Games & Puzzles

TRUTH & FALSEHOOD

In a faraway land there dwelt two races. The Ananias were inveterate liars, while the Diogenes were unfailingly veracious. Once upon a time a stranger visited the land, and on meeting a party of three inhabitants inquired to what race they belonged. The first murmured something that the stranger did not catch. The second remarked, "He said he was an Anania." The third said to the second, "You're a liar!" Now the question is, of what race was this third man?

Mathematical Puzzles

HALVING NUMBERS

When is one half of ten equal to five, but one half of nine equals four, one half of eleven is six and one half of twelve, seven?

Games & Puzzles

ODD ARRANGEMENT

Arrange the nine digits so that the first three shall be ⅓ of the last three; and the central three equal to the difference between the first three and the last three.

Numbers shall be as they are arranged, not totalled.

TELL ME WHY, DAD

Sometimes little boys who ask too many questions tie their parents into mental knots. Take, for example, the young brat who asked his father (a professor of philosophy) if God could do anything. Of course the father said "Yes." "Then can he make a stone so large that he can't roll it?" asked junior. The professor was about to say "Yes" again but remained silent.

Fun With Mathematics

TOO MUCH BEER?

The following is a portion of a report submitted by an investigator for a well-known market analysis agency with standards of accuracy so high that it boasts that an employee's first mistake is his last.

Number of consumers interviewed	100
Number who drink coffee	78
Number who drink tea	71
Number who drink both tea and coffee .	48

Why was the interviewer discharged?

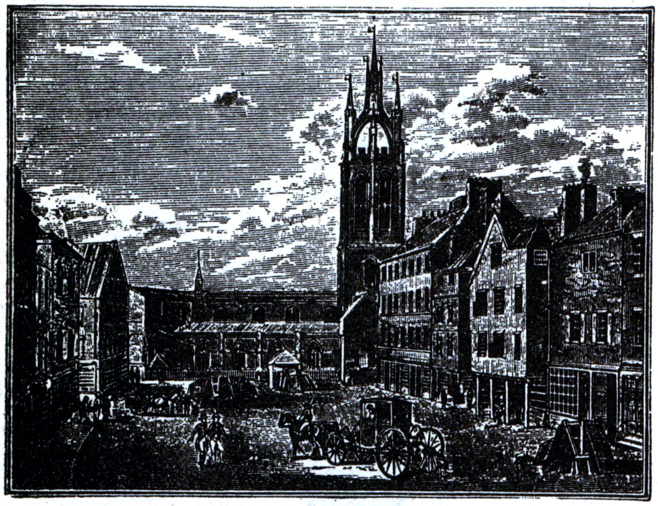
101 Puzzles in Thought and Logic

OLYMPIC WALKERS

A and *B* begin, at 6 a.m. on the same day, to walk along a road in the same direction, *B* having a start of 14 miles, and each walking from 6 a.m. to 6 p.m. daily. *A* walks 10 miles, at a uniform pace, the first day, 9 the second, 8 the third, and so on: *B* walks 2 miles, at a uniform pace, the first day, 4 the second, 6 the third, and so on. When and where are they together?

Pillow Problems

A SMALL TOWN AFFAIR



There is a small town of a few hundred inhabitants of which the following statements are, surprisingly, true:

1) Every man in the town is a perfect logician and is aware that this is true of every other man in the town.
2) Every man in the town knows *all* about the behavior of every woman in the town, with the exception, if he is married, of his own wife. It is taboo for anyone to speak about a woman to her husband.
3) It is an immutable custom (abhorrent to us maybe, but as inevitable as night following day to them) that, when a man discovers that his wife has been unfaithful, he takes her out into the town square that same night, and, on the stroke of midnight, shoots her.
4) There are 40 unfaithful wives in town.

Now, life has been continuing its uneventful course for some time when, one fateful summer's day, June 1st actually, the Mayor summons all the townsmen to a meeting in the town hall. 'I am very sorry to have to tell you this,' he says, 'but there is an unfaithful wife in this town.' The meeting ends and the men disperse. What, if anything, happens, and when? (*Not* an easy problem!)

Games & Puzzles

ONE BAG OR TWO?

There are two bags, one containing a counter, known to be either white or black; the other containing 1 white and 2 black. A white is put into the first, the bag shaken, and a counter drawn out, which proves to be white. Which course will now give the best chance of drawing a white—to draw from one of the two bags without knowing which it is, or to empty one bag into the other and then draw?

Pillow Problems

VERY ODD

There are 8 consecutive odd numbers and when they are multiplied by each other you get 34,459,425.

You'll find them—if you try hard enough.

Mathematical Puzzles and Pastimes

PERSISTENT PADDLER

A woodsman paddling steadily across the still surface of a northern lake saw a magnificent bass break water directly ahead of him. Twelve strokes he counted until his canoe first crossed the ever-widening circle the fish had made, and then twelve more before he broke through the circle on the opposite side. For a time thereafter he sought relief from the pleasant monotony of his journey by calculating how far away the fish had been at the moment it jumped, but it proved too much for him and he soon gave himself up to less specific thoughts.

Can you complete his calculation?

101 Puzzles in Thought and Logic

TYPEWRITER TWISTER

For this puzzle you need to know your typewriter keyboard very well, or otherwise have a typewriter in front of you. Each letter is coded by the row it is in and its position in the row. Thus Q which is the first letter in the first row will be 11, and P, the last letter in that line and the tenth will be 110. D the third letter in the second row will be 23.

A certain word is coded in this manner and then each number from 1 to 10 is replaced by a different number from 1 to 10. Finally the resulting number pairs are converted back, using the same code reversed, into letters. The letters in sequence are: SMFQNEHUX. What was the original word?

Games & Puzzles

SKEW ADDITION

In this addition sum, the powers of 3 have been written under each other, but with each unit digit displaced one to the right compared to the preceding number. Assuming that the powers of 3 are added up forever in this manner, what will the answer be?

Games & Puzzles

```
1
 3
  9
 27
  81
 243
   729
  2187
    6561
    19683
```

NONE GIVEN

If the third is JAS and the fourth is OND, what are the first two?

SEE-SAW

Nine boys, none of whom weighed more than 100 pounds or less than 50 pounds, experimented on a see-saw and found the following lucky thirteen facts:

A. The see-saw balanced because the total weight at each end was exactly equal:
  1. Art and Don with George and Henry
  2. Art and Chuck with Don, George and Ira

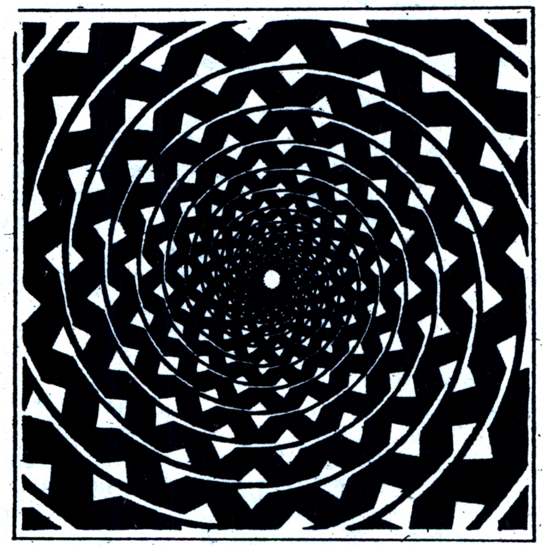
B. These groups were excellent, the differences in weight at the two ends being no more than ten pounds:
  3. Bob and Chuck with Ed and Fred
  4. Chuck and George with Ed and Henry
  5. Don and George with Ed
  6. Art, George & Henry with Chuck and Ed

C. These groups were quite unbalanced, the differences in weight at the two ends being 25 pounds or more:
  7. Ed with Fred
  8. Art with Don
  9. Don with Fred
  10. Chuck with Ira
  11. Bob and Don with Henry and Ira
  12. Chuck and Henry with Fred and Ira
  13. Art, Chuck and Henry with Don, Ed and Fred

How many pounds does each boy weigh? Thinking caps will be required for this one.

Games & Puzzles

Circle or Spiral ?



THREE BAGS

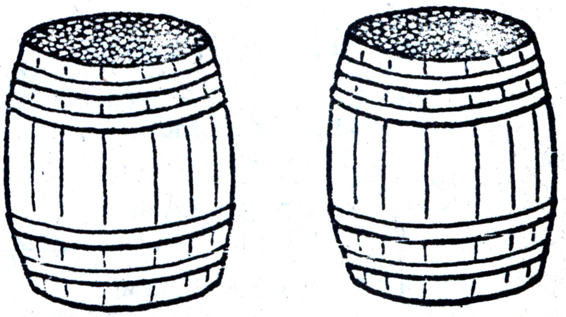
There are 3 bags; one containing a white counter and a black one, another two white and a black, and the third 3 white and a black. It is not known in what order the bags are placed. A white counter is drawn from one of them, and a black from another. What is the chance of drawing a white counter from the remaining bag?

Pillow Problems

BARREL OF FUN

If a millionaire offered you your choice between a barrel filled with half dollars and the same barrel filled with dimes, which would you choose?

Fun With Mathematics



TENNIS AT HILLCREST

Eight men entered the recent tennis tournament at Hillcrest. The tournament was played in three consecutive days, one round per day, and happily no match was defaulted. The first and second round matches were stipulated to be 2 sets out of 3, while the final was 3 sets our of 5. A spectator who was present on all three days reports the following facts:

1. Eggleston never met Haverford.

2. Before play began, Gormley remarked jocularly to Bancroft, "I see that we meet in the finals."

3. Chadwick won a set at love but lost his first match.

4. Altogether 140 games were played, of which the losers won 43.

5. When the pairings were posted, Abercrombie said to Devereaux, "Do you concede, or do you want to play it out?"

6. On the second day, the first-round losers played bridge, and the same table gathered on the third day with Eggleston in place of Abercrombie.

7. Bancroft won 9 games.

8. Franklin won 37 games.

9. The first score of the tournament was a service ace by Gormley, at which Eggleston shouted "Hey, I'm not over there!"

Who won the tournament? Whom did he beat and by what score?

Mathematical Puzzles

THREE COINS

Show the different ways that three coins turn up. There are eight ways in all.

a. What is the chance that the three coins will turn up all heads?

b. What is the chance that the three coins will turn up two heads and one tail?

Invitation to Mathematics

GIGO

What is wrong with the conclusions based on the data given in problems 1 through 4?

1. How many days do you go to school?

	Days
Days in a year:	365
You sleep at least 8 hours per day or ⅓ of the year:	− 122
This leaves:	243
You have 52 Saturdays and 52 Sundays off:	− 104
This leaves:	139
You have summer vacation for three months:	− 90
This leaves:	49
You have Christmas and Easter vacations:	− 19
This leaves:	30
And you spend at least 2 hours each day eating:	− 30
Days left to go to school:	0

2. More people were killed in airplane accidents in 1960 than in 1928. Therefore, it was more dangerous to ride an airplane in 1960 than in 1928.

3. Checkered cows produce 26 percent more milk than other cows. Therefore, checkered cows are the best milkers.

4. There are fewer accidents in France than in Germany. Therefore, it is safer to drive a car in France than in Germany.

Invitation to Mathematics

KRYPTIC KIDS

There are eight brothers and sisters in a family: Georgette, Ulysse, Yvon, Marie, Annette, Roger, Isadore, and Emile. All are less than ten years old. If you represent the age of each child by the first letter of its name, you obtain:

$$\begin{array}{r} YU \\ GUY\,\overline{)MARIE} \\ MARE \\ \hline EE \end{array}$$

If Marie is the youngest of the sisters, what is her age?

150 Puzzles in Crypt-Arithmetic

SIMPLE MULTIPLICATION

Let us take any two numbers + fractions and multiply them together. Then let us take the same two numbers + the equivalent decimals and multiply them again. Naturally you would expect the two answers to be the same. But let's see:

16½	16.5
12½	12.5
32	8.25
16	33.0
8¼ (half of 16½)	165
6¼ (half of 12½)	Answer: 206.25
Answer: 206½	

What has happened to the other ¼?

Fun With Mathematics

COUNTERWEIGHTS

The large flats and other pieces of scenery used in a vaudeville theater are counterweighted by sandbags, so that when they are moved only a small portion of the weight has to be borne by the stagehands.

The theater keeps on hand a set of metal counterweights for occasional use with special pieces. Any or all of the weights can be attached quickly to an elevator rope. There are five weights in the set, so arranged that it is possible to compound any load which is a multiple of 10, from 10 pounds up to the total of all five weights together. The choice of weights is such as to reach the maximum possible total load. What are the several weights?

Mathematical Puzzles

CHINESE MENU

How many guests were present at a Chinese party if every two used a dish for rice between them, every three a dish for broth, every four a dish for meat, and there were 65 dishes altogether?

The Math Entertainer

LIFT SIMULATION

Write a program to simulate a lift, capable of stopping at 4 floors. The lift is controlled by 2 buttons on the middle 2 floors (requesting a lift up and a lift down), a single button on the top and bottom floors and a set of 4 buttons in the lift itself corresponding to the 4 floors.

The lift is controlled in such a way that the buttons within the lift take precedence over those on each of the floors. However, the lift can be made to stop at a floor as a result of a request there, if the lift is passing thru that floor in the right direction or alternatively if there is no request from within the lift itself.

Requests for the lift are queued and, apart from the provisos given above, are dealt with on a first come first served basis.

The program simulating the lift allows the user to press any of the buttons (by typing at the teleprinter), just before it is about to reach a floor, whether or not it is due to stop there. If there are no requests at all, the program terminates. Output from the program should indicate where the lift is, the direction it is moving and where it stops.

Computer Programming Problems

BATTER UP

How many possible batting orders are there for a baseball team of 9 players?

WEIGHT WATCHERS

The fat men in a club outnumber the thin men by sixteen. Seven times the number of fat men exceeds nine times the number of thin men by thirty-two. Find the number of fat and thin men in the club.

The Math Entertainer

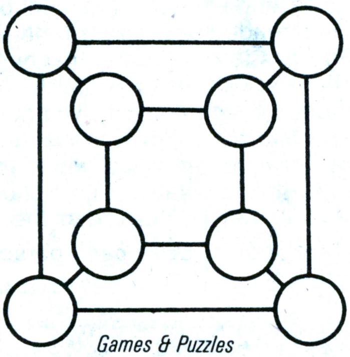
SEESAW

Three brothers go to a playground to play on the seesaw. The teeter board has a fixed seat at each end, 5 feet away from the trestle on which the board swings. When Alfred and Bobby take seats, Charles, who weighs 80 pounds, balances them by sitting on Alfred's side 21 inches away from the trestle. When Charles sits on a seat, it takes both his brothers to balance him, Alfred in the other seat and Bobby one foot nearer the center. Now if Bobby takes Alfred's place, where must Alfred sit to balance Charles?

Mathematical Puzzles

SQUARE EIGHTEEN

How can the numbers 1 to 8 be placed, one in each of these circles, so that the sums of the numbers at the corners of each of the five areas, and the sum of the four numbers on the outside are all equal to 18 and the 7 and 8 are not in two corners of the same area or square?



Games & Puzzles

FAST FREIGHT

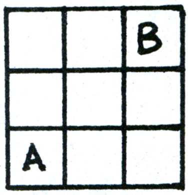
If it takes twice as long for a passenger train to pass a freight train after it first overtakes it as it takes the two trains to pass when going in opposite directions.

How many times faster than the freight train is the passenger train?

101 Puzzles in Thought and Logic

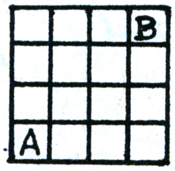
A SIMPLE PUZZLE

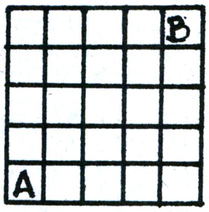
Try this set of simple problems with your students and see if they can draw general conclusions about how the problem would be solved on different game boards. Draw a 3" by 3" board with A and B marked as follows:

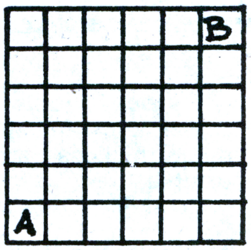


Put a penny on A. What is the smallest number of moves it would take the penny to go from A to B if it could move 1 square either horizontally or vertically per turn? The answer which is 4 is pretty easy to discover. Here's a slightly more difficult question however: How many different paths are there that takes 4 moves to get from A to B? The answer is 8.

Now try to answer the same questions on a 4" x 4", 5" x 5" and a 6" x 6" board.







Now fill in the following chart:

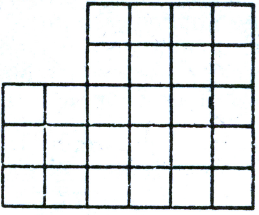
Board	*Smallest number of moves from A to B*	*Number of smallest paths*
3x3	4	8
4x4		
5x5		
6x6		

Do you see any patterns? Can you describe them in simple words? Can you also describe them in mathematical symbols?

Gamesmag

SQUARES

How many squares can you find in the diagram below?



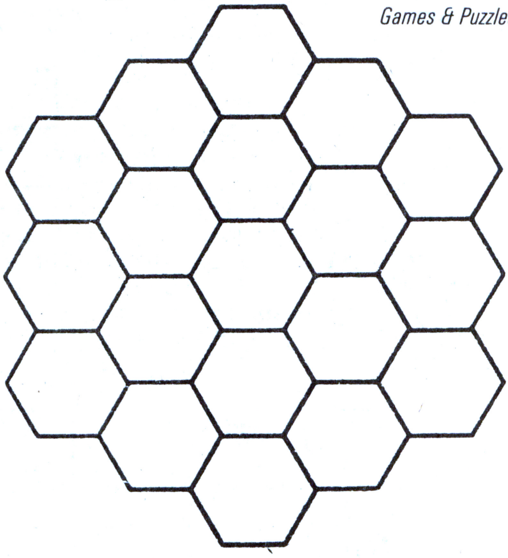
LOGIC ONLY

$$\begin{array}{r} A\ B\ C\ D\ E \\ \times\ 4 \\ \hline E\ D\ C\ B\ A \end{array}$$

1 TO 19

Can you place the numbers 1 to 19 in the honeycomb's nineteen cells in such a way that there is a difference of at least 4 between any one cell and its neighboring cells?

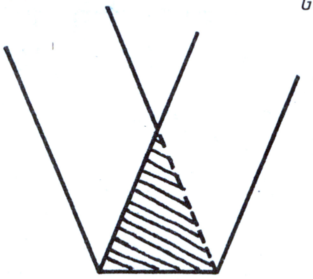
Games & Puzzles



A DOUBLE STRIP

A strip of paper is folded once, as in the figure. At what angle must it be folded so that the double thickness area (shaded) is as small as possible.

Games & Puzzles



ALPHABET SOUP

The word "crypt-arithmetic" was first introduced by M. Vatriquant writing under the pseudonym "Minos". In the May, 1931 issue of *Sphinx,* a Belgian magazine of recreational mathematics, he proposed this problem with these remarks:

"Cryptographers, to hide the meaning of messages, put figures in places of letters. By way of reprisal, we will replace each digit of the following problem with a distinct letter."

$$\begin{array}{r} ABC \\ \times\, DE \\ \hline FEC \\ DEC \\ \hline HGBC \end{array}$$

150 Puzzles in Crypt-Arithmetic

Although the word was new, the type of puzzle was older. The earliest one in my collection is from the *Strand Magazine* for July, 1924:

TWO X TWO = THREE

ADJACENT SQUARES

49 and 169 are both squares. They have something else in common—they can both be formed by placing two squares adjacent to each other, using 4 and 9, and 16 and 9 respectively.

What are the next two squares with this property?

Games & Puzzles

SIX BOYS

The same six boys are to sit around a table for lunch. How many different arrangements can be made of the order in which they are to sit?

The Math Entertainer

AND WHO'S THE TRACK COACH?

In the Stillwater High School the economics, English, French, history, Latin, and mathematics classes are taught, though not necessarily respectively, by Mrs. Arthur, Miss Bascomb, Mrs. Conroy, Mr. Duval, Mr. Eggleston, and Mr. Furness.

The mathematics teacher and the Latin teacher were roommates in college.

Eggleston is older than Furness but has not taught as long as the economics teacher.

As students, Mrs. Arthur and Miss Bascomb attended one high school while the others attended a different high school.

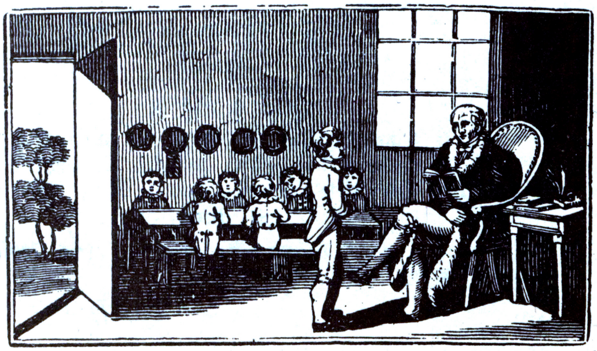
Furness is the French teacher's father.

The English teacher is the oldest of the six both in age and in years of service. In fact he had the mathematics teacher and the history teacher in class when they were students in the Stillwater High School.

Mrs. Arthur is older than the Latin teacher.

What subject does each person teach?

101 Puzzles in Thought and Logic



SANTA CLAUS

If the value of CHRISTMAS is 110, what is the most likely value of NEW YEAR?

LAST DIGIT ONLY

What is the last digit in 7^{1000}? Try this in BASIC (remember: not the entire number, just the last digit).

SHIRT SALE

"I made a smart move marking down those shirts from $2.00," remarked Mr. Gaberdine to his wife. "We have disposed of the entire lot."

"Good!" said Mrs. Gaberdine. "How much profit did you made?"

"We haven't figured it yet, but the gross from the sale was $603.77."

"Well, how many shirts did you sell?"

Let the reader answer the question.

Mathematical Puzzles

NOT QUITE USELESS

The well-known firm of Shyster, Shyster, Shyster, Shyster & Sons is in trouble. The calculating machines they use to work out their fees are badly broken down. One will only add, another will only subtract, the third will only multiply, and the fourth will do nothing but divide. Which machine will be of most use to the poor clerks, who are totally useless at mental arithmetic?

Games & Puzzles

NEXT IN SEQUENCE PLEASE

1. Ten letters—what is the eleventh?

NWHOIIEIIE?

2. The answer to this one is far out, outa sight, just too much.

MVEMJSUN?

Games & Puzzles

THE THREE BEGGARS

A charitable lady met a poor man to whom she gave one cent more than half of the money she had in her purse. The poor fellow, who was a member of the United Mendicants' Association, managed, while tendering his thanks, to chalk the organization's sign of "a good thing" on her clothing. As a result, she met many objects of charity as she proceeded on her journey.

To the second applicant she gave two cents more than half of what she had left. To the third beggar she gave three cents more than half of the remainder. She now had one penny left.

How much money did she have when she started out?

Puzzles of Sam Lloyd

MARK TWAIN'S SUGGESTION

Is it true (as Mark Twain once suggested) that with two dice one is twice as likely to throw 7 as to throw 10?

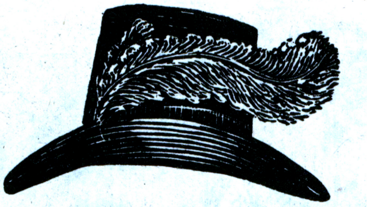
Floodgate Scandal

THE OVERWORKED LIBRARIAN

Our local librarian has been very busy. On Monday she cataloged only some of the new books received. Tuesday she received as many new books as were uncataloged on Monday, and cataloged 10. Wednesday she received 12 more than on Monday, and cataloged as many as she had done on that day. Three times as many books arrived on Thursday as she had cataloged on Wednesday, and 8 were cataloged. On Friday, 6 books arrived and 12 fewer were cataloged than were received on Wednesday. On Saturday she was able to catalog the outstanding 16 books as the library was closed. How many books arrived on Monday?

Games & Puzzles

COMING TO TOWN



Uncle Reuben and Aunt Cynthia came to town to shop. Reuben bought a suit and hat for $15. Cynthia paid as much for her hat as Reuben did for his suit; then she spent the rest of their money for a new dress.

On the way home, Cynthia called Reuben's attention to the fact that his hat cost $1 more than her dress. Then she added: "If we had divided our hat money differently so that we bought different hats, mine costing 1 and ½ times the cost of yours, then we each would have spent the same amount of money."

"In that case," said Uncle Reuben, "how much would my hat have cost?"

Can you answer Reuben's question and also tell how much money the couple spent altogether?

Puzzles of Sam Lloyd

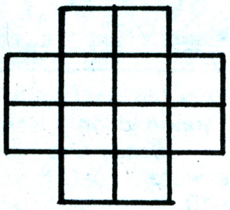
TO THE MOVIES

An old-time movie house charged admission prices of 25 cents for an adult and 10 cents for a child. If the cashier in the box-office after closing time counted the ticket stubs and found that they totaled 385 while the money amounted to $62.65—how many children had entered?

Mathematical Puzzles and Pastimes

1 TO 12

Place the integers 1 through 12 in the 12 spaces in the grid below so that each of the two rows, two columns, and five squares that can be formed with 4 numbers in each have a total of 26. How many *fundamentally* different ways can the numbers be arranged in the grid (don't count rotations or mirror images)? Can you find the arrangement in which no two consecutive numbers are next to each other vertically, horizontally or diagonally?



BUT SUMMER SUN . . .

Each different letter in this sum stands for a different digit from 1 to 9. What is the original addition sum?

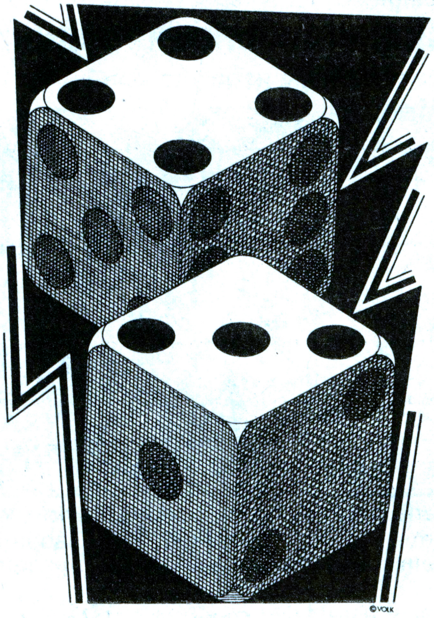
```
SPRING
 RAINS
 BRING
 GREEN
------
PLAINS
```

Games & Puzzles

ERIC'S FAIR DICE

'Let's decide like this,' said Eric, 'I've got two dice here. We each choose a number and I'll throw the dice until the total on the two uppermost faces is a number one of us has chosen, and he will go first.' 'That's no good,' said David, 'some totals are more likely than others, and anyway there are too many of us.' 'Wrong on both counts,' replied Eric, grinning, 'these are special dice. Every different pair of faces gives a different total, and the totals start from 1 and go up consecutively. OK?' 'I'll believe that when I see the dice,' said David. He looked at them and Eric was quite right: they had what numbers on their faces?

Games & Puzzles



SUMMARIZING

The numbers 1 to 16 have already been placed in a square array. Your puzzle is to turn each horizontal and vertical column into a correct sum by placing between each pair of numbers, vertically and horizontally, a plus, minus or equals sign using eight of each.

1	2	3	4
8	7	6	5
9	10	11	12
16	15	14	13

Games & Puzzles

HARDWARE SUPPLIES

A man went into a hardware store to buy something for his house. He asked the clerk the price, and the clerk replied, "The price of one is twelve cents, the price of 30 is twenty-four cents, and the price of 144 is thirty-six cents." What did the man want to buy?

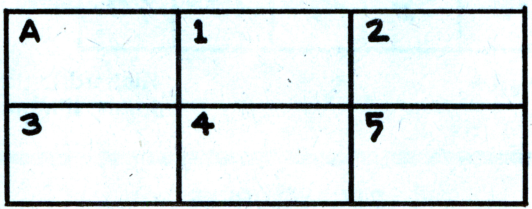
EVEN ADDITION

The number 156 is the sum of the first 12 even numbers. Curiously enough, if it is beheaded we get 56 which is the sum of the first 7 even numbers, and beheading again, 6 is the sum of the first two. There is one other three-figure number with the same property and just one 4-figure number. What are they?

Games & Puzzles

MAP FOLDING

Here is a teaser for anyone who has ever had trouble refolding a road map. Try solving it in your head. If our six-section map below is folded so that section A is on top facing you, which of the other five sections would it be possible to have on the bottom facing the other way?



TWO COINS

In how many ways can two coins turn up? The possible ways are heads-heads, heads-tails, tails-heads, tails-tails.

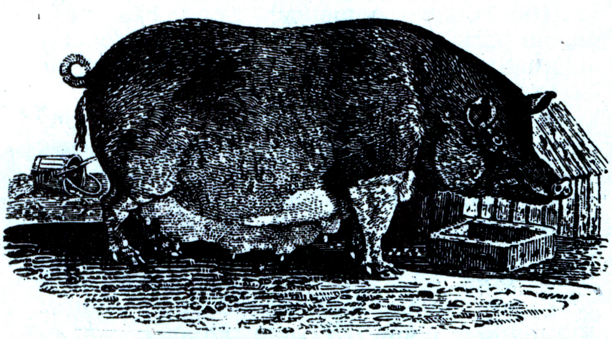
a. What is the chance that the two coins will turn up two heads?
b. What is the chance that the two coins will turn up one head and one tail?

Invitation to Mathematics

A POWER PROBLEM

The integer 844,596,301 is the 5th power of what number?

TOM THE PIPER'S SON



According to Mother Goose, Tom the Piper's son stole the pig and away he run. When Tom started after the pig, he was standing 250 yards due south of the pig. Both began running at the same time and ran with uniform speeds. The pig ran due east. Instead of running north-east on a straight line, Tom ran so that at every instant he was running directly toward the pig.

Assuming that Tom ran 1 and ⅓ times faster than the pig, how far did the pig run before he was caught? The simple rule for solving this type of problem is based on elementary arithmetic, but will doubtless be new to most of our puzzlists.

Puzzles of Sam Lloyd

SUMS IN CODE

This coded sum uses all four rules of arithmetic. Can you supply the figures? Each different letter stands for a different digit from 0 to 9.

```
     P Y F
         P  (multiplication)
   -------
   G P G Y
       F A  (addition)
   -------
   G P T A
       L E  (subtraction)
   -------
F ) G F Z G (division)
   -------
     T G A
```

Games & Puzzles

GOT THE POINT?

The arithmetic in line A below is obviously incorrect, but with the insertion of two dots (as decimal points) in line B, the line A equation is made correct. Now see if you can insert two dots somewhere in the third line to correct the faulty arithmetic:

A. (72 x 3) — 5 = 166
B. (7.2 x 3) — 5 = 16.6
Problem: (51 — 3) x 2 = 34

A DEAL IN CANDY

Three boys received a nickel each to spend on candy. The stock offered by the candy store comprised lollipops at 3 for a cent, chocolate bonbons at 4 for a cent, and jujubes at 5 for a cent. Each boy made a different selection, but each spent his entire 5 cents and returned with just 20 pieces of candy. What were their selections?

Mathematical Puzzles

ONE PILE

This purely numerical game has been traced back to remote antiquity, and probably it antedates the games of position, such as tic-tac-toe.

A number of pebbles or counters of any description is massed in one pile. The two players draw alternately from the pile, the object being to gain the last counter.

If it were permitted to seize the whole pile, the first player would of course win; if the draw were limited to one counter at a turn, the result would depend upon whether the number in the pile were originally odd or even. Therefore, a minimum draw of one counter is set, with a maximum greater than one.

Suppose the limits are 1 to 3 counters. Then if a player finds just 4 counters left in the pile, he loses. Whatever he takes, his opponent can take the remainder. It is readily seen that the number 4 is a critical one because it is the sum of the minimum and maximum limits of the draw.

In order to leave his opponent with 4 counters to draw from, a player must previously have left him 8. Whether he then drew 1, 2, or 3, it was possible to reduce the pile to 4. Evidently the series of winning combinations, each of which is a number to be left in the pile for the opponent to draw from, is simply the muliples of 4.

If we denote "a winning combination" by w, and the least and most that may be drawn at a turn by a and m respectively, then

$$w = (a + m)n$$

where n is any integer. This formula is quite general, and is independent of the number of counters originally in the pile. If this number is of form w, the first player loses; if it is not, he wins by reducing it to w.

TO LEAVE THE LAST

The game can also be played with the object of forcing one's opponent to take the last counter. I leave it to the reader to write the formula for w in this case.

Mathematical Puzzles

BASIC

In base 10, 88^2 is 7744 (two different pairs of like integers). What is the smallest number squared in base 3, which will produce the same pattern?

In base 8?

Write a computer program to identify all such numbers up to base 20.

QUICKIES

1. What is the next letter in this sequence NNNENEENEEE?
2. If 6*8 = 8, 7*13 = 9, 10*15 = 12½, 10*16 = 14, what is 9*11?
3. If COS scores 0 and MEW scores 4, how many will HANKY score?

Games & Puzzles

FIND THE PATH

Can you trace a path from the lower left to upper right square (vertical, horizontal, or diagonal moves are OK) that totals exactly 100?

→ End

17	41	29	3	2
81	4	22	11	8
62	1	56	42	15
35	16	13	14	21
4	39	5	19	56

↗ Start

Richard Latta
Joliet, Illinois

DECAY CURVES

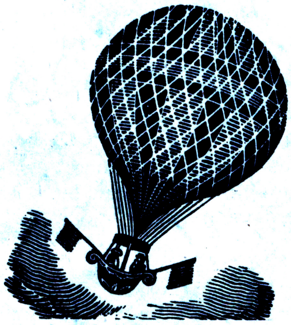
1000 coins are tossed. Those which fall tails are removed and the remainder tossed again. Those which fall tails are removed and so on. A tedious experiment to perform but one which is easy to simulate. A program written to do this should record the result of each tossing as a point on a graph. The graph would have 'coins remaining' across the page and 'throw number' down the page.

Changing the probability from ½ to an input value λ, the simulation reflects radioactive decay. λ is the probability that an atom of a particular element decays in unit time and the graph shows the number of atoms which are not decayed at each time.

The program can be improved by allowing any decayed atoms to decay again as a second radioactive material with a probability λ. Output is then to consist of a combined graph showing the number of atoms of each material.

Computer Programming Problems

FROM A BALLOON



If a stone is dropped from a balloon on a still day, does it fall directly below the balloon, or to the west of it, or to the east?

Floodgate Scandal

The Mystic

by Anthony Dickins

FROM time immemorial, the number 7 seems to have possessed certain mysterious, or even 'magic', properties. Probably this originated with the calculations of the calendar in the early prehistoric period when chief priests, magicians, astrologers and adepts of all kinds all over the world were engaged in observing the motions of the Sun, the Moon and the Stars, carefully recording their results. The 365 days of the solar year were (nearly) divisible into 52 'weeks' of 7 days each (with only one day left remaining to account for). It is possible that at one time and in some places the year was divided into 12 months of 30 days each, the five 'extra' days being treated as 'dark' days around the winter solstice from 21st December (the shortest day) to the 25th when it was clear beyond doubt that the sun was not dead but was being reborn. But with the four solstices and equinoxes of the year occurring at regular intervals, it seems most likely that the year was first divided into four 'quarters', each having 91 days. This would leave only one day per year unaccounted for. The number 91 has only two factors, 7 and 13, and this may well have been the influence that led to establishing 13 weeks of 7 days each in the 91-day quarter. If so, this, too, probably accounts for the very deep-rooted superstition about the number 13, which is found in cultures at great distances apart. It is notable that 7 and 13 are the two numbers most heavily charged with the aura of superstition and mysticism.

Mathematically, the number 7 has several remarkable properties. Here are some well-known ones, together with some less well-known and some quite new.

The fraction one-seventh expressed in decimal form is 0.142857142857... The six figures 142857 recur infinitely. Treating 142857 as a six digit number, and setting out a multiplication table by the first six numbers, we get this curious result:

142857 x 1 = 142857
142857 x 3 = 428571
142857 x 2 = 285714
142857 x 6 = 857142
142857 x 4 = 571428
142857 x 5 = 714285

These are the six 'cyclic permutations' of the number 142857 which is the only number less than a million whose cyclic permutations are all multiples of itself. It will also be noticed that there is a palindromic effect in the digits, columns 1 to 6 from left to right being identical with rows 1 to 6 from top to bottom.

Another curious feature of this cyclic number is that the dirits form a cyclic set, not only in our usual scale 10, but also all other scales from base-9 inclusive, upwards. Obviously, this does not hold for the binary scale, base-2, and other lower-scales where only digits of lesser value than 9, 8, 7, 6, etc., may be used.

In passing, it may be noted that the next known 'cyclic' number of this sort has 18 digits — 052, 631, 578, 947, 368, 421. The leader who wishes to verify this will have a pleasant surprise when he multiplies this number by 2, 3, 4, . . . 10, and sets the products down in rows beneath the original number just as was done with 142857 in the previous paragraph.

Reverting to our original decimal expression for the fraction one-seventh (0.142857), if we divide this by 7, we get another recurring decimal, this time of 42 (7×6) digits:

$$0.\underline{0}20408\underline{1}63265\underline{3}06122\underline{4}48979$$
$$\underline{5}91836\underline{7}34693\underline{8}77551\ \ldots$$

In this number, there are 7 six-digit sequences, each beginning with a different digit, 0134578. Notice that the ten different digits are represented as evenly as possible in a 42-digit number. There are five 3's, five 6's, and four each of the other digits.

Another 'sevenly' number is 5040, or factorial 7. (Factorial seven, written 7!, is equal to 7x6x5x4x3x2x1). 5040 was noticed by Plato as being remarkable for having a very large number of divisors. It is divisible by all but one of the first 12 numbers, and all but six of the first 24 numbers. How quickly can you identify these exceptions?

The following arrangements show properties that are not unique to 7, but which it shares, to a greater or lesser extent, with other numbers.

(1234567 x 9) + 8 = 11111111
12345679 x 63 = 777777777
1 + 2 + 3 + 4 + 5 + 6 + 7 + 6 + 5 + 4 + 3 + 2 + 1 = 7^2

Here is an even more complex peculiarity:

111111 = 7 x 15873
222222 = 7 x 31746
333333 = 7 x 47619
444444 = 7 x 63492
555555 = 7 x 79365
666666 = 7 x 95328
777777 = 7 x 111111
888888 = 7 x 126984
999999 = 7 x *142857* and we arrive back at this unique number.

The number SEVEN is found in various non-thematical connections, both in classical times, and also in more recent history. There were the SEVEN Wonders of the World; there were the SEVEN Sages of Greece, the SEVEN Argive heroes, the SEVEN Hills of Rome. There were also the SEVEN Seas (the Arctic, the Antarctic, the North Pacific, the South Pacific, the North Atlantic, the South Atlantic, and the Indian Ocean).

In mystical and religious connections, the number SEVEN has played a large part, too. In the Book of Revelations, there are SEVEN Stars and SEVEN Golden Candlesticks, which represent SEVEN Churches; a lamb with SEVEN horns and SEVEN eyes which are the SEVEN spirits of God, which are also SEVEN lamps of fire; SEVEN seals of the Book; SEVEN angels; SEVEN trumpets; SEVEN thunders; SEVEN vials; SEVEN kings; and a dragon with SEVEN heads. The number 7 also seems to have held a rather special importance for St. John. The Dance of the SEVEN Veils and the SEVEN Pillars of Wisdom should also be mentioned.

One common theory for the construction of the Universe in classical times, when the Sun was supposed to be of the Heavenly Bodies circling the Earth, included seven concentric circles or 'rings' perpetually revolving round the Earth somewhere many miles up in the sky — hence the term 'Seventh Heaven'.

In the twentieth century, the famous physicist, Niels Bohr, who contributed so much towards the mathematics and the understanding of the structure of the atom, postulated a nucleus having, as it were, seven shells, analogous to the seven notes of the musical scale or the seven planets of antiquity.

The Seven Days of the Creation and the Seven Deadly Sins are further religious uses. In Isaiah, there are the Seven Gifts of the Holy Ghost; and in Matthew (chapter 15), there are seven loaves to feed the four thousand and seven baskets of crumbs left over. The constellation of stars, the Pleiades, is also known as the Seven Sisters; and the Seven Sisters are also the chalk cliffs just east of Cuckmere Haven, Sussex. In popular legend, there are the Seven-leagued boots of Hop o' my Thumb, and the man with Seven Wives going to St. Ives. We shouldn't forget, either, the supposed seven years' bad luck that accompanies the breaking of a mirror.

One way and another, the number 7 has had a good innings in mathematics and history. To finish off with, here is one final fraction:

$$1234567654321 = \frac{7777777 \times 7777777}{1+2+3+4+5+6+7+6+5+4+3+2+1}$$

Reprinted with permission from Games & Puzzles. Copyright 1975 by Edu-Games (U.K.) Ltd.

'Not every problem is one to be solved by computer programming.'

TURNING A PUZZLE INTO A LESSON

Eugene D. Homer
C. W. Post College, Greenvale, NY

The second problem in the feature column "Puzzles and Problems For Fun", *Creative Computing,* 1, 4, (May-June, 1975) proved to be an ideal nucleus for class discussion in a course in advanced programming, although the results were not what, I presume, the author intended.

The problem was stated thusly:

> *"Mr. Karbunkle went to the bank to cash his weekly paycheck. In handing over the money, the cashier, by mistake, gave him dollars for cents and cents for dollars.*
>
> *"He pocketed the money without examining it and spent a nickel on candy for his little boy. He then discovered the error and found he possessed exactly twice the amount of the check.*
>
> *"If he had no money in his pocket before cashing the check, what was the exact amount of the check? One clue: Mr. Karbunkle earns less than $50 a week."*

I assume the intent of the author was to have readers write a computer program to solve the problem by trial and error. My intent was to show the class how analysis of the problem before coding could simplify the program. I would like to share this lesson and the resulting conclusion with the readers of *Creative Computing.*

Our first step was to state whatever relationships we could from the problem in mathematical form.

Let D be the integer number of dollars and C the integer number of cents on Mr. Karbunkle's paycheck. The total amount printed on his check, expressed in cents, is:

$$A = 100D+C \tag{1}$$

Since the teller reversed D and C, the amount of cash Mr. Karbunkle received, again expressed in cents, is:

$$R = 100C+D \tag{2}$$

We are told that

$$R-5=2A \tag{3}$$

Substituting Equations 1 and 2 in Equation 3, we obtain:

$$100C+D-5=2(100D+C),$$

which can be simplified to:

$$199D=98C-5, \tag{4}$$

We have one equation, in two integer unknowns, which does look like a problem for trial and error solution. If we were to code at this point, we might come up with something like this, remembering that D is less than 50:

```
   INTEGER D,C
   DO 1 D = 1, 49
   N = 199*D
   DO 1 C = 1, 100
   L = 98*(C—1)—5
   IF (N—L) 1,2,1
2  A = (100*D+C)/100
   WRITE (5,3) A
   CALL EXIT
1  CONTINUE
   WRITE (5,4)
   CALL EXIT
3  FORMAT (F7.2)
4  FORMAT (1X,'NO SOLUTION')
   END
```

Although this looks like a fairly simple program, I pointed out that it would require a maximum of 4,900 repetitions of the main loop. (In the following, it becomes convenient to measure interatives by the number of times the IF statement is executed.) In view of this large amount of computation we agreed that the analyst should attempt two things:

a) Reduce the amount of computation in the loop, and
b) Reduce the number of times the program must loop.

Tackling the first idea, the class came up with such suggestions as replacing lines 2 and 3 of the above program with the line:

```
DO 1 N = 199, 9751, 199
```

This led to a similar discussion about simplifying lines 4 and 5 of the above program. In this round, it became apparent that C must be greater than 1, since for C=1, L = —5, and L would never be equal to N. It also became apparent that our inner loop could be terminated as soon as L became greater than N. We now had our program down to something like this:

```
   INTEGER D,C
   DO 1 N=199, 9751, 199
   DO 5 L=93, 9697, 98
   IF (N—L) 5, 2, 1
2  A=(100*D+C)/100.
   WRITE (5, 3) A
   CALL EXIT
5  CONTINUE
1  CONTINUE
   etc.
```

We had reduced our maximum number of executions of the IF statement by almost one-half (to 2,499 executions) and had removed all arithmetic calculation from the loops.

We were still not happy with the program, since the inner loop, on L, was too repetitive. We saw that if a particular value of L was less than a particular value of N, there was no need to try that value of L again for the next value of N. This led us to the removal of the inner loop altogether:

```
      INTEGER D,C
      L = 93
      DO 1 N = 199, 9751, 199
      IF (N-L) 1, 2, 5
    2 A = (100*D+C)/100.
      WRITE (5, 3) A
      CALL EXIT
    5 L=L+98
    1 CONTINUE
      etc.
```

A few "runs" by hand of this program indicated that we would try only two values of L for each value of N, reducing the maximum number of IF statement executions to 98.

Before we left this approach, we took another look at Equation 4. For any non-negative integer value of C, the right side of the equation will be odd. Therefore the term 199 D must be odd, and therefore D must be odd. Thus, D may assume only the values 1,3,5,...,49, and N = 199 D will increase in increments of 2(199)=398. Our last program, then, can have its DO statement changed to

```
DO 1 N = 199, 9751, 398,
```

resulting in another halving of the iterations.

It was now time to take another tack. I reminded the class of last week's work with modular numbers, and showed them that Equation 4 satisfied the first definition of a modular number,

$$N = q \cdot m + r,$$

but failed the second definition,

$$0 < r < (m-1)$$

where $N = 199D$
$q = C$
$m = 98$
$r = -5$

However, we could rewrite Equation 3 as:

$$199D = 98C - 98 - 5 + 98$$

or

$$199D = 98(C-1) + 93$$

which satisfied both definitions of modular numbers with

$N = 199D$
$q = C-1$
$m = 98$
$r = 93$

We could then write, from the familiar expression

$$\begin{aligned} r &\equiv N \bmod m, \\ 93 &\equiv 199D \bmod 98 \qquad (5) \end{aligned}$$

Our strategy then could be to use the MOD function as we increment N. If we find a value of N satisfying Equation 5, we can solve for

$$D = \frac{N}{199}$$

and, by rewriting equation 4

$$C = \frac{N+5}{98}$$

Our new program follows. Note that we have also dropped the integer declaration as being unnecessary.

```
      DO 1 N = 199, 9751, 398
      IR = MOD(N,98)
      IF (IR-93) 1, 2, 1
    2 D = N/199
      C = (N+5)/9800.
      A = D+C
      WRITE (5, 3) A
      CALL EXIT
    1 CONTINUE
      WRITE (5,4)
      CALL EXIT
    3 FORMAT (F7.2)
    4 FORMAT (1X,'NO SOLUTION')
      END
```

This looked like a reasonable program, requiring only 24 repetitions of the IF statement, maximum. We set out to run it by hand, with these results:

N	IR
199	3
597	9
995	15
1393	21

At this point the class saw that as N increased by 199, IR increased by 6, and that it might be possible to "figure out" when IR would hit 93.

It took only a few minutes to work out the fact that

for $N = 199 + 398i;\ i = 0,1,2,\ldots$
$IR = 3 + 6i$

If IR is to be equal to 93:

$$93 = 3 + 6i$$

or,

$$i = 15$$

This would occur when

$$N = 199 + 398(15) = 6169,$$

and

$$D = \frac{6169}{199} = 31,$$

$$C = \frac{N+5}{98} = 63,$$

$$A = \$31.63$$

To be sure, we checked with equations 2 and 3:

$$R = \$63.31$$

$$\$63.31 - \$.05 = \$63.26 = 2(\$31.63).$$

Our "program" now has reduced to:

```
      WRITE (5,5)
      CALL EXIT
    5 FORMAT (1X,'$31.63')
      END
```

We spent about two hours going over this puzzle. While much of our work was useless in terms of the final solution, the class did learn some valuable lessons from the discussion. They learned that careful analysis of a problem can lead to a startling reduction in the amount of computing to be done. They learned that it pays to run through a program "by hand" a few times to discover hidden relationships. Finally, they learned that not every problem presented in a computer-oriented environment is a problem to be solved by computer programming.

Non-Usual Mathematics for Computer Solution

James Reagan
Stevenson High School, Sterling Heights, Michigan

Introduction

Mathematics instruction generally proceeds sequentially and deductively. This instructional procedure creates some misconception of the mathematics. Mathematics is not totally deductive logic; the deductive proof of any hypothesis is developed after one has become quite certain that the conjecture is true. One investigates enough specific cases to become somewhat sure that the observed cases generalize or that the proper limits on the conjecture have been found. Thus, there is a contradiction between the mathematics in its instruction and mathematics in its historical development. In formal instruction in mathematics the discovery of the theorems, rules, and properties are taught as though they were bestowed upon man as were the two tablets containing the Ten Commandments; the time and effort expended are seldom discussed. Most mathematics courses offer the student the deductive process in developing the material when, historically, the deductive process was employed late in the development of the topic.

Because of mathematics instruction's dependence upon deductive development, certain topics fall before or after certain other topics; and mathematics instruction has become characterized by its sequential approach. It is true that there are certain foundations upon which some topics rest; these pre-requisites are necessary for the development of the vocabulary and the organization of latter theorems. With the use of computers in many schools, some of the latter topics can be studied out of sequence.

Agreed, there is what might be called mathematics sophistication before one can *master* certain topics, but how much mathematics sophistication is required to *understand* and *appreciate* the material? It is this writer's experience that students in the secondary school can investigate topics and solve problems prior to the traditional time location of the topic or problem in the instructional sequence. Many topics commonly deferred to the college curriculum are suitable and interesting for the secondary school student.

What follows in this series are examples of such problems that have been studied and solved by high school students in computer programming classes at Stevenson High School and many other high schools having computer access.

Infinitely Many Primes

Background of the Problem

The great mathematician of the third century B. C., Euclid, proved that there are infinitely many primes. Euclid's proof leads to interesting problems some 2000 years later.

First the proof and then the problems.

The proof is by *reductio ad absurdum,* an indirect proof.

Suppose there are finitely many prime numbers. Then, these n primes can be listed in order.

$$2, 3, 5, \ldots, P_n.$$

Form a number N by adding 1 to the product of the n primes:

$$N = 2*3*5*\ldots*P_n + 1.$$

Either N is prime or N is composite. Each of these results for N leads to a contradiction that P_n is the largest prime.

First, if N is prime, then it is clearly greater than P_n and P_n is not the greatest prime.

Second, suppose N is composite. It has a prime factor p. This prime factor p cannot be one of the primes $2, 3, 5, \ldots, P_n$, since dividing each of the primes in the list into N leaves a remainder of 1. Thus, p must be a prime greater than P_n.

Therefore, there are infinitely many prime numbers. QED.

Statement of the Problem

The creation of the number N in the proof by Euclid leads to many interesting questions.

Create a set of numbers by the recursive definition:

$$\begin{aligned} P_1 &= 2 \\ P_2 &= 3 \\ P_3 &= P_1*P_2 + 1 = 2*3 + 1 = 7 \\ P_4 &= P_1*P_2*P_3 + 1 = 2*3*7 + 1 = 43 \\ &\vdots \\ P_{n+1} &= P_1*P_2*P_3*\ldots*P_n + 1. \end{aligned}$$

Are each of the numbers in the set prime?

If some number in the list is not prime, is a prime factor of it greater than the preceding number in the list? For example, if P_6 is composite, is a prime factor of P_6 greater than P_5?

These same questions apply to a second set that can be created by subtracting 1 instead of adding 1 to the previous list product.

Create a second set as follows:

$$\begin{aligned} P_1 &= 2 \\ P_2 &= 3 \\ P_3 &= P_1*P_2 - 1 = 2*3 - 1 = 5 \\ &\vdots \\ P_{n+1} &= P_1*P_2*\ldots*P_n - 1. \end{aligned}$$

Answer the same questions as with the first set.

Finally, create a third set and investigate.

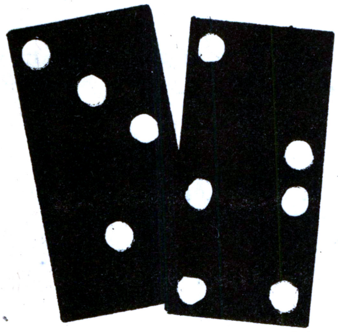
$$\begin{aligned} P_1 &= 2 \\ P_2 &= 3 \\ P_3 &= 5 \\ P_4 &= P_1*P_2*P_3 + 1 = 2*3*5 + 1 = 31 \\ &\vdots \\ P_{n+1} &= P_1*P_2*\ldots*P_n + 1. \end{aligned}$$

Hints

1. One of the constraints on the computations is the number of significant digits of the computing machine. The numbers in each of the sets become large rapidly and can soon overflow the significant digit capacity of the machine. An extended precision routine may be needed to investigate far into the sets. [See "Computing Factorials — Accurately" by Walter Koetke, *Creative Computing* Vol 1, No 3, pp 9–11.]

2. To save on variable storage during execution of the program, instead of using high dimensioned vectors, it might be helpful to create a file in which to store new numbers in the set and from which to read out previcus numbers.

In computer play, the aggressive method of play is significantly better than the defensive.



SIMULATED STRATEGIES OF GAME PLAYING

by Dr. S. Reisman
IBM Canada, Ltd. Laboratory

Game theory enables one to classify competitive games according to characteristics of rules of play of the game. Consequently, the game of chess is described as a *two-person, zero sum* game of *perfect* information. This classification describes a competitive situation between two players in which the total game situation is open to observation to both players. Both players have opposite interests in the outcome and one player's win is the other's loss. In addition, the game can be terminated in a finite number of moves. Games of perfect information differ from games of imperfect information by the fact that, in the latter, there is no best strategy. However, in the former, sometimes called strictly determined games, the player with the best strategy will win regardless of his opponent's play.

The game of Draw and Match Dominoes is classified as a *two-person, zero sum* of *imperfect* information and according to game theory there is no best strategy of play which can be employed.

It was hypothesized while this might in fact be true, techniques of simulating cognitive processes might be used to determine if better strategies do exist. Consequently, player protocols were gathered from opponents while they played a simplified version of Draw and Match Dominoes. An analysis of the protocols indicated that human players use a combination of a number of components of play to form their playing strategy. These components are described as: (1) the defensive component; (2) the aggressive component; and, (3) the statistical component.

The defensive component is characterized by a player's being more apt to make assumptions about his opponent's game situation and as a result attempting to block his every move. The aggressive component is different in that the player using it makes no assumptions about his opponent's situation and instead makes his move only on the basis of his own known situation with the objective of playing his longest chain of tiles. The statistical component is used rarely, and only as a last means of decision making if the other two components are unsatisfactory. It is characterized by the player's counting the various tiles already played and making decisions as to the likelihood of matches being drawn from the bank.

In an effort to determine the validity of this classification system an interactive Dominoes-playing program was written in the list processing language IPL-V.* The program was altered so that the strategy employed in a game could be varied to combinations of the above components. Results of the computer play indicated that the classification system is, in fact, an operational one.

In order to determine if there are optimal strategies of play, the interactive program was altered to allow the play of the game to be between two computer programs rather than between a human and a computer, as in the original version. In this way, one program using one type of strategy could be played against a program of another strategy. The programs were each loaded onto a different interactive terminal and the output of one terminal's program was used as input to the terminal with the second player-program.

In this attempt to determine a better strategy, one program was set to use only the defensive component, and the other, only the aggressive component. The results of a number of games played in this manner indicate that the aggressive component of play is significantly better than the defensive component. An analysis of the games played indicated the reasons for this. The strictly defensive player makes assumptions concerning his opponent's situation while ignoring his own game situation and the harm he may be doing to himself by blindly attacking his opponent.

The typical human player does not rely on only one component of play, but both makes assumptions about his opponent and considers his own situation. For this reason, a better strategy would probably consist of a combination of these components in a particular game situation. Although the search for a better strategy of play in the game of Dominoes is not of earth-shaking importance, the potential spinoff of the techniques used and results obtained may be of value in other situations of a competitive nature.

---

*For a complete description see: Dominoes—A Computer Simulation of Cognitive Processes, *Simulation and Games*, Vol. 3, No. 2, pp. 155-164.

Wumpus 2

by Gregory Yob

Hark!! The weary Wumpus hunter, wan from 50 days in the Terminal Caverns, exhausted and with all of his arrows expended — — — (A groaning Teletype roars at a sleepy student. Maps litter the floor covered with circles and integers. With callused fingers, the immortal Wumpus player looks up with bloodshot eyes and implores: "How do I get out of here?")

I suspected that the dodecahedron may prove a bit boring after a few thousand games, so I wrote Wumpus 2 to extend your pleasure. Some of the more mathematical minded may have noticed there are lots of ways to link caves with three tunnels apiece. Some of these patterns are topologically interesting . . . .

Wumpus 2 is the same old Wumpus* in different settings — including those of your own design. As you play in the different caves, you will notice that the game changes in difficulty and strategy. Now to a description of the various caves in Wumpus 2.

CAVE 0 (Dodecahedron)

This is the same old Wumpus with which you are familiar.

CAVE 1 (Mobius Strip)

Since my original vision was topological, here is the first wonder of topology, the Mobius strip. Take a strip of paper, give it a half-twist and join the ends into a loop. The result has just one side and one edge (if you disbelieve, take a pencil and go around the thing).

A perceptive player will note that the placement of the pits influence the game. Two pits placed just right (around 5% of the games have this) will force a detour back around the strip in certain cases. Getting around is slower than in Cave 0, but it is easier to search the place.

CAVE 2 (String of Beads)

See the diagram for this one. Here, placement of the pits will often make parts of the caves inaccessible except by bat-express. (Can you see why?) Play in this cave is frustrating until you have gone to the trouble of making a reference map; otherwise you keep coming back to your starting point. (Look at the diagram and see how this may be so.)

CAVE 3 (Hex Network)

This is my attempt at a torus (doughnut). If you can visualise a hexagon net like a honeycomb or a tile floor and stretch it onto a doughnut, you've got it!! The drawing tries to show this, but if you prefer, think of it as a complicated molecule of some sort. Play in this one is very similar to CAVE 0.

CAVE 4 (Dendrite)

Up to now, each tunnel leads to another cave and only one tunnel connects a pair of caves. This need not be a strict rule and the next two caves illustrate variations on this. The dendrite is a branching pattern like a tree or a plant. At the ends of the plant are "leaves" which are caves leading to themselves or multiple tunnels. This cave is especially susceptible to severance by pits and getting stuck in corners near the wumpus. A nice thing is that you often will know exactly where the Wumpus is when you come near him.

CAVE 5 (one way streets)

This is the extreme example of all tunnels are one-way. You will find that getting about this cave is like travel in Los Angeles — much going to get to the neighbor's house. If you overshoot, you must travel all the way around, just like missing a freeway offramp.

CAVE 6 (Do Your OWN)

Draw up a map of caves, each cave with tunnels GOING TO three (exactly three) caves (same or different). Then the computer will ask you for the numbers of the destination tunnels for each of the 20 caves in Wumpus. When you have it entered, play Wumpus on your own caves. Let me know of your favorite ones, and your most frustrating ones!!!

If you are a programming fiend, how would you arrange a Wumpus game with lots of caves and tunnels — and legitimate moves may include changing how the caves connect to each other?? Send me any versions which may happen.

WUMPUS 3

Around the PCC center, a Howard lives. He enjoys suggesting "improvements" to my games and I enjoy telling him how useless his ideas are. One day, he got around me and came up with Wumpus 3, with earthquakes, bat migrations and the incredible "tumareo" (don't let your fantasy get away from you now!!).

Have a look at the run and see if you like it. My personal opinion is that the changes and reshufflings happen too often for comfort — what do you think?

We didn't have room for Wumpus 3. Send to Gregory for a tape.

FINIS

In any case, Wumpus has spawned several versions and spread about the computer games-dom really nicely. For myself, the soul of the game is in the idea and fun of it rather than the program or the computer which hosts it. I feel that all really good games will turn programmers on enough for them to write it for their system from the idea alone and encourage games writers to think carefully on the art and esthetics of their games before writing a line of code.

WUMPUS TAPES, ETC.

I can be found at:

Gregory Yob
PO Box 354
Palo Alto, Calif. 94301

Paper tapes of Wumpus, Wumpus 2 and Wumpus 3 are available and cost $5.00 each.

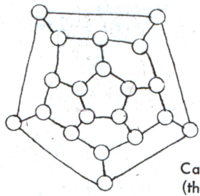
May your arrows remain straight. —Gregory Yob.

Sample run, listings, and cave diagrams on the next 2 pages.

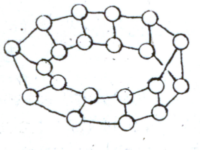
---

* Wumpus 1 appeared in Creative Computing, Vol 1, No 5 (Sep-Oct '75).

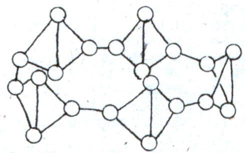
THE CAVES OF WUMPUS 2



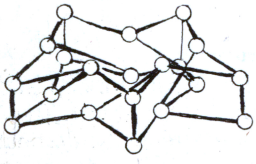
Cave 0
(the usual)



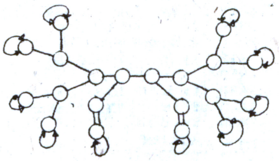
Cave 1
(Mobius Strip)



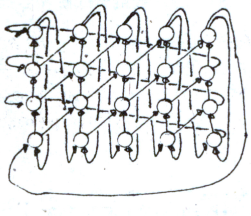
Cave 2
(String of Beads)



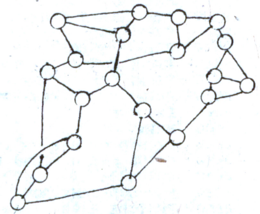
Cave 3
(Toroidal Hex Net)



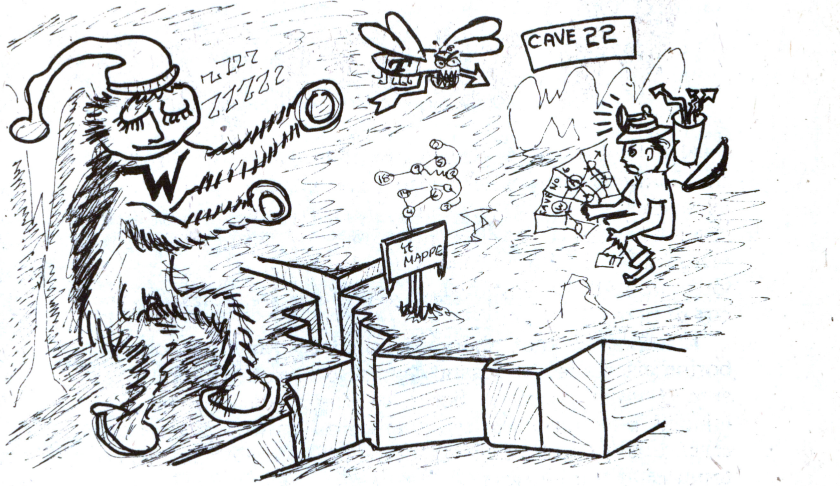
Cave 4
(Dendrite)



Cave 5
(One Way Only)



Cave 6
(Anything You Like)



WUMPUS 2 SAMPLE RUN

We didn't print the instructions on the run. They are in the listing in Statements 700-1210.

```
HUNT THE WUMPUS

I SMELL A WUMPUS!
I FEEL A DRAFT
BATS NEARBY!
YOU ARE IN ROOM  11      TUNNELS LEAD TO  9       10       12

SHOOT OR MOVE? ?S

NO. OF ROOMS (1-5)? ?3

ROOM #?9

ROOM #?10

ROOM #?11


OUCH! ARROW GOT YOU!
HA HA HA - YOU LOSE!
PLAY AGAIN? ?Y

SAME SET-UP? ?Y

HUNT THE WUMPUS

I SMELL A WUMPUS!
I FEEL A DRAFT
BATS NEARBY!
YOU ARE IN ROOM  11      TUNNELS LEAD TO  9       10       12

SHOOT OR MOVE? ?M

WHERE TO? ?9


ZAP--SUPER BAT SNATCH! ELSEWHEREVILLE FOR YOU!
... OOPS! BUMPED A WUMPUS!

I SMELL A WUMPUS!
I FEEL A DRAFT
YOU ARE IN ROOM  12      TUNNELS LEAD TO  10       11       13

SHOOT OR MOVE? ?M

WHERE TO? ?13


YOU ARE IN ROOM  13      TUNNELS LEAD TO  12       14       15

SHOOT OR MOVE? ?S

NO. OF ROOMS (1-5)? ?2

ROOM #?12

ROOM #?11


AHA! YOU GOT THE WUMPUS!  HE WAS IN ROOM 11
HEE HDE HEE - THE WUMPUS'LL GETCHA NEXT TIME!!
```

WUMPUS 2 LISTING

PART 1

```
0001  REM: WUMPUS II BY GREGORY YOB REV MAY 1975
0002  REM: GREGORY YOB PO BOX 354 PALO ALTO, CALIF 94301
0003  REM: PHONE (415) 326-4039
0004  REM: MODIFIED FROM HP BASIC VERSION. NOTE THAT:
0005  REM: THE EXPRESSION I$(1,1) IS REPLACED BY I$(1:1)
0006  REM: IN THIS VERSION.
0010  DIM I$(6)
0020  REM- WUMPUS VERSION II
0050  PRINT "INSTRUCTIONS? ";
0060  INPUT I$
0070  PRINT
0090  IF I$(1:1) <> "Y" THEN 130
0100  GOSUB 700
0110  REM-CHOOSE & SET UP CAVE
0120  DIM S(20,3)
0130  GOSUB 2530
0140  DEF FNA(X)=INT(20*RND(0))+1
0150  DEF FNB(X)=INT(3*RND(0))+1
0160  DEF FNC(X)=INT(4*RND(0))+1
0170  REM-LOCATE L ARRAY ITEMS
0180  REM-1-YOU,2-WUMPUS,3&4-PITS,5&6-BATS
0190  DIM L(6)
0200  DIM M(6)
0210   FOR J=1 TO 6
0220   L(J)=FNA(0)
0230   M(J)=L(J)
0240   NEXT J
0250  REM-CHECK FOR CROSSOVERS (IE L(1)=L(2),ETC)
0260   FOR J=1 TO 6
0270    FOR K=J TO 6
0280    IF J=K THEN 300
0290    IF L(J)=L(K) THEN 210
0300    NEXT K
0310   NEXT J
0320  REM-SET# ARROWS
0330  A=5
0340  L=L(1)
0350  REM-RUN THE GAME
0360  PRINT "HUNT THE WUMPUS"
0370  REM-HAZARD WARNINGS & LOCATION
0380  GOSUB 1230
0390  REM-MOVE OR SHOOT
0400  GOSUB 1400
0410  GOTO 0 OF 430,470
0420  REM-SHOOT
0430  GOSUB 1550
0440  IF F=0 THEN 400
0450  GOTO 490
0460  REM-MOVE
0470  GOSUB 2150
0480  IF F=0 THEN 380
0490  IF F>0 THEN 540
0500  REM-LOSE
0510  PRINT "HA HA HA - YOU LOSE!"
0520  GOTO 550
0530  REM-WIN
0540  PRINT "HEE HDE HEE - THE WUMPUS'LL GETCHA NEXT TIME!!"
0550   FOR J=1 TO 6
0560   L(J)=M(J)
0570   NEXT J
0580  PRINT "PLAY AGAIN? ";
0590  INPUT I$
0600  PRINT
0620  IF I$(1:1) <> "Y" THEN 3310
0640  PRINT "SAME SET-UP? ";
0650  INPUT I$
0660  PRINT
0680  IF I$(1:1) <> "Y" THEN 130
0690  GOTO 330
0700  REM-INSTRUCTIONS
0710  PRINT "WELCOME TO WUMPUS II"
0720  PRINT "THIS VERSION HAS THE SAME RULES AS 'HUNT THE WUMPUS'."
0730  PRINT "HOWEVER, YOU NOW HAVE A CHOICE OF CAVES TO PLAY IN."
0740  PRINT "SOME CAVES ARE EASIER THAN OTHERS. ALL CAVES HAVE 20"
0750  PRINT "ROOMS AND 3 TUNNELS LEADING FROM ONE ROOM TO OTHER ROOMS."
0760  PRINT "THE CAVES ARE:"
0770  PRINT "  0  -  DODECAHEDRON      THE ROOMS OF THIS CAVE ARE ON A"
0780  PRINT "              12-SIDED OBJECT, EACH SIDE FORMING A PENTAGON."
0790  PRINT "              THE ROOMS ARE AT THE CORNERS OF THE PENTAGONS,"
0800  PRINT "              EACH ROOM HAVING TUNNELS LEADING TO 3 OTHER ROOMS.
                                      "
0810  PRINT "  1  -  MOBIUS STRIP      THIS CAVE IS TWO ROOMS"
0820  PRINT "              WIDE AND 10 ROOMS AROUND (LIKE A BELT)"
0830  PRINT "              YOU WILL NOTICE THERE IS A HALF-TWIST"
0840  PRINT "              SOMEWHERE."
0850  PRINT
```

WUMPUS 2

LISTING

PART 2

```
0860  PRINT "  2  -  STRING OF BEADS     FIVE BEADS IN A CIRCLE."
0870  PRINT "          EACH BEAD IS A DIAMOND WITH A VERTICAL"
0880  PRINT "          CROSS-BAR.  THE RIGHT & LEFT CORNERS LEAD"
0890  PRINT "          TO NEIGHBORING BEADS.(THIS ONE IS DIFFICULT"
0900  PRINT "          TO PLAY)"
0910  PRINT
0920  PRINT "  3  -  HEX NETWORK      IMAGINE A HEX TILE FLOOR. TAKE"
0930  PRINT "         A RECTANGLE WITH 20 POINTS (INTERSECTIONS)"
0940  PRINT "         INSIDE (4X4). JOIN RIGHT & LEFT SIDES TO MAKE A"
0950  PRINT "         CYLINDER. THEN JOIN TOP & BOTTOM TO FORM A"
0960  PRINT "         TORUS (DOUGHNUT)."
0970  PRINT "         HAVE FUN IMAGINING THIS ONE!!"
0980  PRINT
0990  PRINT "  CAVES 1-3 ARE REGULAR IN THE SENSE THAT EACH ROOM"
1000  PRINT "  GOES TO THREE OTHEER ROOMS & TUNNELS ALLOW TWO-"
1010  PRINT "  WAY TRAFFIC.  HERE ARE SOME 'IRREGULAR' CAVES:"
1020  PRINT
1030  PRINT "  4  -  DENDRITE WITH DEGENERACIES   PULL A PLANT FROM"
1040  PRINT "          THE GROUND.  THE ROOTS & BRANCHES FORM A"
1050  PRINT "          DENTRITE - IE, THERE ARE NO LOOPING PATHS"
1060  PRINT "          DEGENERACY MEANS  A) SOME ROOMS CONNECT TO"
1070  PRINT "          THEMSELVES AND B) SOME ROOMS HAVE MORE THAN"
1080  PRINT "          ONE TUNNEL TO THE SAME OTHER ROOM IE, 12 HAS"
1090  PRINT "          TWO TUNNELS TO 13."
1100  PRINT
1110  PRINT "  5  -  ONE WAY LATTICE     HERE ALL TUNNELS GO ONE"
1120  PRINT "         WAY ONLY.  TO RETURN, YOU MUST GO AROUND THE"
1130  PRINT "         CAVE (ABOUT 5 MOVES)."
1140  PRINT
1150  PRINT "  6  -  ENTER YOUR OWN CAVE     THE COMPUTER WILL ASK"
1160  PRINT "         YOU THE ROOMS NEXT TO EACH ROOM IN THE CAVE."
1170  PRINT "         FOR EXAMPLE:"
1180  PRINT "           ROOM # 1     ?2,3,4     - YOUR REPLY OF 2,3,4"
1190  PRINT "             MEANS ROOM 1 HAS TUNNELS GOING TO ROOMS:"
1200  PRINT "             2, 3 & 4."
1210  PRINT "  HAPPY HUNTING!"
1220  RETURN
1230  REM-PRINT LOCATION & HAZARD WARNINGS
1240  PRINT
1250   FOR J=2 TO 6
1260    FOR K=1 TO 3
1270    IF S(L(1),K)#L(J) THEN 1340
1280    GOTO J-1 OF 1290,1310,1310,1330,1330
1290    PRINT "I SMELL A WUMPUS!"
1300    GOTO 1340
1310    PRINT "I FEEL A DRAFT"
1320    GOTO 1340
1330    PRINT "BATS NEARBY!"
1340    NEXT K
1350   NEXT J
1360  PRINT "YOU ARE IN ROOM ";L(1);
1370  PRINT " TUNNELS LEAD TO ";S(L,1);S(L,2);S(L,3)
1380  PRINT
1390  RETURN
1400  REM-CHOOSE OPTION
1410  GOTO 1450
1420  PRINT "ERROR   ";
1430  INPUT Z9
1440  PRINT "";
1450  PRINT "SHOOT OR MOVE? ";
1460  INPUT I$
1470  PRINT
1490  IF I$(1:1) <> "S" THEN 1520
1500  O=1
1510  RETURN
1520  IF I$(1:1) <> "M" THEN 1420
1530  O=2
1540  RETURN
1550  REM-ARROW ROUTINE
1560  F=0
1570  REM-PATH OF ARROW
1580  DIM P(5)
1590  GOTO 1630
1600  PRINT "ERROR    ";
1610  INPUT Z9
1620  PRINT "";
1630  PRINT "NO. OF ROOMS (1-5)? ";
1640  INPUT J9
1650  PRINT
1670  IF J9<1 OR J9>5 OR INT(J9) <> ABS(J9) THEN 1600
1680   FOR K=1 TO J9
1690   PRINT "ROOM #";
1700   INPUT P(K)
1710   PRINT
1730   IF P(K)>0 AND P(K)<21 AND INT(P(K))=ABS(P(K)) THEN 1780
1740   PRINT "ERROR    ";
1750   INPUT Z9
1760   PRINT "";
1770   GOTO 1690
1780   NEXT K
1790  PRINT
1800  REM-SHOOT ARROW
1810  A=A-1
1820  A9=L(1)
1830   FOR K=1 TO J9
1840    FOR K1=1 TO 3
1850    IF S(A9,K1)=P(K) THEN 1990
1860    NEXT K1
1870   REM-NO TUNNEL FOR THE ARROW
1880   A9=S(A9,FNB(1))
1890   GOTO 2000
1900   NEXT K
1910  PRINT "MISSED"
1920  REM-MOVE WUMPUS
1930  GOSUB 2070
1940  REM-AMMO CHECK
1950  IF A>0 THEN 1970
1960  F=-1
1970  RETURN
1980  REM-SEE IF ARROW IS AT L[1] OR L[2]
1990  A9=P(K)
2000  IF A9 <> L(2) THEN 2040
2010  PRINT "AHA! YOU GOT THE WUMPUS!  HE WAS IN ROOM";L(2)
2020  F=1
2030  RETURN
2040  IF A9 <> L(1) THEN 1900
2050  PRINT "OUCH! ARROW GOT YOU!"
2060  GOTO 1960
2070  REM-MOVE WUMPUS ROUTINE
2080  K=FNC(0)
2090  IF K=4 THEN 2140
2100  L(2)=S(L(2),K)
2110  IF L(2)#L THEN 2140
2120  PRINT "TSK TSK TSK- WUMPUS GOT YOU!"
2130  F=-1
2140  RETURN
2150  REM- MOVE ROUTINE
2160  F=0
2170  GOTO 2210
2180  PRINT "ERROR   ";
2190  INPUT Z9
2200  PRINT "";
2210  PRINT "WHERE TO? ";
2220  INPUT L
2230  PRINT
2240  IF L<1 OR L>20 OR INT(L) <> ABS(L) THEN 2180
2250   FOR K=1 TO 3
2260   REM-CHECK IF LEGAL MOVE
2270   IF S(L(1),K)=L THEN 2350
2280   NEXT K
2290  IF L=L(1) THEN 2350
2300  PRINT "NOT POSSIBLE - ";
2310  INPUT Z9
2320  PRINT "";
2330  GOTO 2210
2340  REM-CHECK FOR HAZARDS
2350  L(1)=L
2360  REM-WUMPUS
2370  IF L#L(2) THEN 2430
2380  PRINT "... OOPS! BUMPED A WUMPUS!"
2390  REM-MOVE WUMPUS
2400  GOSUB 2080
2410  IF F=0 THEN 2430
2420  REM-PIT
2430  IF L <> L(3) AND L <> L(4) THEN 2480
2440  PRINT "YYYIIIIEEEE . . . FELL IN A PIT"
2450  F=-1
2460  RETURN
2470  REM-BATS
2480  IF L <> L(5) AND L <> L(6) THEN 2520
2490  PRINT "ZAP--SUPER BAT SNATCH! ELSEWHEREVILLE FOR YOU!"
2500  L=FNA(1)
2510  GOTO 2350
2520  RETURN
2530  REM-SELECT CAVE
2540  GOTO 2580
2550  PRINT "ERROR   ";
2560  INPUT Z9
2570  PRINT "";
2580  PRINT "CAVE #(0-6)? ";
2590  INPUT N
2600  PRINT
2620  IF N<0 OR N>6 OR INT(N) <> ABS(N) THEN 2550
2630  GOSUB N+1 OF 2650,2730,2810,2890,2970,3050,3130
2640  RETURN
2650  REM-DODECAHEDRON
2660  RESTORE 2590
2670  DATA 2,5,8,1,3,10,2,4,12,3,5,14,1,4,6
2680  DATA 5,7,15,6,8,17,1,7,9,8,10,18,2,9,11
2690  DATA 10,12,19,3,11,13,12,14,20,4,13,15,6,14,16
2700  DATA 15,17,20,7,16,18,9,17,19,11,18,20,13,16,19
2710  GOSUB 3240
2720  RETURN
2730  REM-MOBIUS STRIP
2740  RESTORE 2750
2750  DATA 20,2,3,19,1,4,1,4,5,2,3,6,3,6,7
2760  DATA 4,5,8,5,8,9,6,7,10,7,10,11,8,9,12
2770  DATA 9,12,13,10,11,14,11,14,15,12,13,16,12,16,17
2780  DATA 14,15,18,15,18,19,16,17,20,2,17,20,1,18,19
2790  GOSUB 3240
2800  RETURN
2810  REM-STRING OF BEADS
2820  RESTORE 2830
2830  DATA 2,3,20,1,3,4,1,2,4,2,3,5,4,6,7
2840  DATA 5,7,8,5,6,8,6,7,9,8,10,11,9,11,12
2850  DATA 9,10,12,10,11,13,12,14,15,13,15,16,13,14,16
2860  DATA 14,15,17,16,18,19,17,19,20,17,18,20,1,18,19
2870  GOSUB 3240
2880  RETURN
2890  REM-HEX NET ON TORUS
2900  RESTORE 2910
2910  DATA 6,10,16,6,7,17,7,8,18,8,9,19,9,10,20
2920  DATA 1,2,15,2,3,11,3,4,12,4,5,13,5,6,14
2930  DATA 7,16,20,8,16,17,9,17,18,10,18,19,6,19,20
2940  DATA 1,11,12,2,12,13,3,13,14,4,14,15,5,11,15
2950  GOSUB 3240                        4,14,15,5,11,15 ← BUG!!
2960  RETURN                                   ↖This is the culprit
2970  REM- DENDRITE W/ DEGENERACIES
2980  RESTORE 2990
2990  DATA 1,1,5,2,2,5,3,3,6,4,4,6,1,2,7
3000  DATA 3,4,7,5,6,10,8,9,9,8,8,10,7,9,11
3010  DATA 10,13,14,12,13,13,11,12,12,11,15,16,14,17,18
3020  DATA 14,19,20,15,17,17,15,18,18,16,19,19,16,20,20
3030  GOSUB 3240
3040  RETURN
3050  REM-ONE WAY LATTICE
3060  RESTORE 3070
3070  DATA 5,4,8,1,5,6,2,6,7,3,7,8,8,9,12
3080  DATA 5,9,10,6,10,11,7,11,12,12,13,16,9,13,14
3090  DATA 10,14,15,11,15,16,16,17,20,13,17,18,14,18,19
3100  DATA 15,19,20,1,4,20,1,2,17,2,3,18,3,4,19
3110  GOSUB 3240
3120  RETURN
3130  REM- INPUT OWN CAVE
3140   FOR J=1 TO 20
3150   PRINT "ROOM #";J;
3160   INPUT S(J,1),S(J,2),S(J,3)
3170    FOR K=1 TO 3
3180    IF S(J,K)>0 AND S(J,K)<21 AND INT(S(J,K))=ABS(S(J,K))
3190    PRINT "***** ERROR!!!!!"
3200    GOTO 3150                                  → THEN 3210
3210    NEXT K
3220   NEXT J
3230  RETURN
3240  REM-INPUT CAVE
3250   FOR J=1 TO 20
3260    FOR K=1 TO 3
3270    READ S(J,K)
3280    NEXT K
3290   NEXT J
3300  RETURN
3310  END
```

Another new game from Creative Computing . . . .

WAR 3

WAR3 is a version of the tried and true Gunner game, occasionally called the "Battleship" game. It can be played by two or three players, or one playing oneself (still very exciting). The original version was written by Mike Forman, revisions by M. E. Lyon and Brian West of Morse High School, San Diego.

The game is rather straight forward, and generates excitement, with a great amount of vicarious aggression.

A suggested sequence to use with WAR3 might be:

1. Study the trajectories of objects noting:
    a. Angles of Launch
    b. Velocity
    c. Effect of gravity upon projectiles
    d. Distance, and its relation to velocity and angle of launch.
2. Discuss the observations of trajectories of objects such as a ball, a pebble (small), etc.
3. Play WAR3 with the aid of the computer.
4. Challenge those who have played WAR3 to develop a similar game. Maybe, modify WAR3 to allow four or more players.

Listing and sample run of WAR3 on next page.

Another new game from Creative Computing . . . .

DR. Z

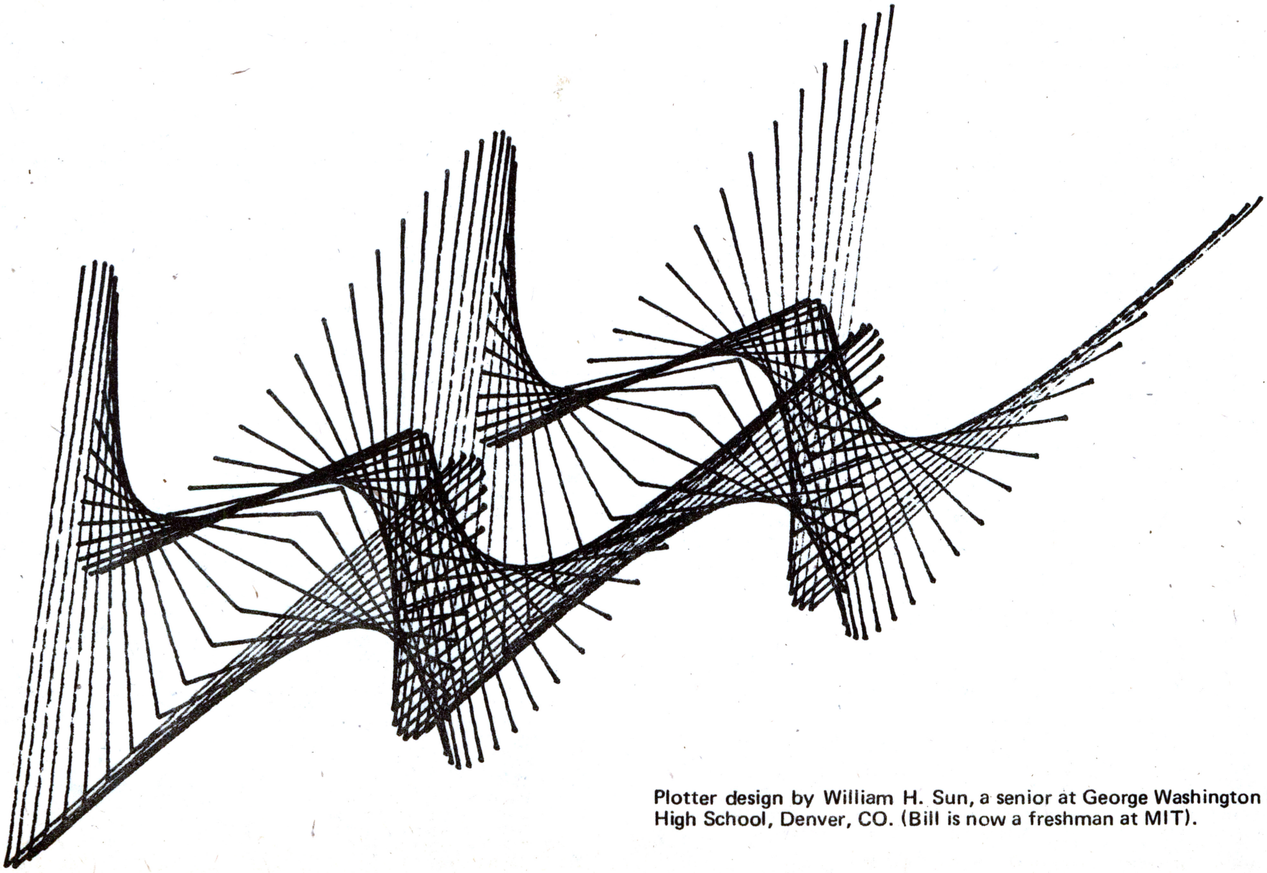
Using DR.Z your computer "interacts" with you in true Rogerian form, never making a value judgment of your response.

DR.Z is multi-lingual and "professional confidence" is guaranteed, especially with a video display terminal. However, if you have a printer, try employing a unique language known only to you and DR.Z.

If you would prefer to employ DR.Z in an educational mode, you might consider the following sequence of activities.

1. Discuss communication, exploring:
    a. The role of spoken language
    b. The role of written language
    c. Non-verbal language
        1. facial expressions
        2. posture of body
        3. hand gestures
2. Experience a session with DR.Z.
3. Develop computerizations of other "purely human" situations.

Listing and sample run of Dr. Z are 1 page over.



Plotter design by William H. Sun, a senior at George Washington Sr. High School, Denver, CO. (Bill is now a freshman at MIT).

WAR 3

LISTING

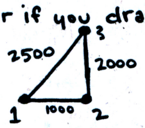
```
WAR3

10  DIM A$[3]
20  T=0
30  REM 'WAR 3'·  ORIGINAL 8K MODV FOCAL BY MIKE FORMAN
40  REM TSS/8 BASIC IV VERSION BY M E LYON JR 1972
50  REM H-P 2000 VERSION BY BRIAN WEST 1975
60  DIM V[3],X[3],P[3],R[3,3]
70  MAT V=ZER
80  MAT X=ZER
90  MAT P=ZER
100  MAT R=ZER
110  DATA 1,2,2,3,3,1,1,3,3,2,2,1,2,3,3,1,1,2,0
120  PRINT "THIS IS THE BASIC VERSION OF 'WAR3'·  TWO OR THREE MAY PLAY"
130  PRINT "DO YOU NEED INSTRUCTIONS";
140  INPUT A$
150  IF A$="YES" THEN 1310
160  PRINT ""
170  PRINT "NO· OF PLAYERS";
180  INPUT N
190  IF N=2 THEN 240
200  IF N=3 THEN 270
210  PRINT "ERROR--TWO OR THREE PLAYERS·"
220  PRINT
230  GOTO 160
240  N1=1
250  PRINT ""
260  GOTO 290
270  N1=N
280  PRINT ""
290  FOR J=1 TO N1
300  READ A,B
310  PRINT "DISTANCE (FT·) ";A;" TO ";B;
320  INPUT R[A,B]
330  R[B,A]=R[A,B]
340  NEXT J
350  PRINT ""
360  RESTORE
370  IF N=2 THEN 460
380  FOR J=1 TO N
390  READ A,B,C,D,E,F
400  IF R[A,B]<R[C,D]+R[E,F] THEN 440
410  PRINT "ERROR--ILLEGAL TRIANGLE·  RE-ENTER RANGES·"
420  RESTORE
430  GOTO 290
440  NEXT J
450  PRINT
460  FOR J=1 TO N
470  PRINT "MUZZLE VELOCITY (FT·/SEC·) OF ";J;
480  INPUT V[J]
490  NEXT J
500  PRINT ""
510  FOR J=1 TO N
520  X[J]=V[J]↑2/32
530  NEXT J
540  FOR A=1 TO N
550  FOR B=1 TO N
560  IF X[A]>R[A,B] THEN 610
570  PRINT "ERROR--";A;" CANNOT REACH ";B
580  PRINT "WHAT IS THE MUZZLE VELOCITY OF ";A;
590  INPUT V[A]
600  GOTO 510
610  NEXT B
620  NEXT A
630  N1=N
640  PRINT ""
650  PRINT ""
660  PRINT "ROUND ";T+1
670  PRINT
680  FOR M=1 TO N
690  IF N=3 THEN 750
700  C=1
710  IF M <> 1 THEN 730
720  C=2
730  PRINT "PLAYER ";M;" SHOOTING AT";C
740  GOTO 890
750  IF P[M]=12 THEN 1280
760  PRINT "PLAYER ";M;" SHOOTING AT";
770  INPUT C
780  IF C=1 THEN 830
790  IF C=2 THEN 830
800  IF C=3 THEN 830
810  PRINT "ERROR--PLAYERS DESIGNATED 1, 2, 3·"
820  GOTO 760
830  IF C <> M THEN 860
840  PRINT "ERROR--CANNOT SHOOT SELF·"
850  GOTO 760
860  IF P[C] <> 12 THEN 890
870  PRINT "ERROR-- ";C;" IS DEFUNCT"
880  GOTO 760
890  PRINT "FIRING ANGLE";
900  INPUT A3
910  IF A3<0 THEN 940
920  IF A3>180 THEN 940
930  GOTO 970
940  PRINT "ERROR--FIRED INTO GROUND·  ";M;" NOW DEFUNCT·"
950  P[M]=12
960  GOTO 760
970  IF A3<90 THEN 1000
980  PRINT "ERROR--FIRED WRONG WAY, LOSE SHOT·"
990  GOTO 760
1000  Z=SIN(A3*3·49064E-02)*V[M]↑2/32
1010  X=(R[M,C]/1000*RND(0))-(R[M,C]/1000*RND(0))
1020  D=X+Z
1030  D1=R[M,C]*·05
1040  IF D<D1 THEN 1080
1050  IF ABS(D-R[M,C])<D1 THEN 1110
1060  IF D<R[M,C] THEN 1140
1070  IF D>R[M,C] THEN 1160
1080  PRINT "  TOO CLOSE - ";M;" IS DEFUNCT·"
1090  P[M]=12
1100  GOTO 1180
1110  PRINT "  A HIT - ";C;" IS DEFUNCT·"
1120  P[C]=12
1130  GOTO 1180
1140  PRINT "  YOU UNDERSHOT BY ";ABS(D-R[M,C]);" FEET·"
1150  GOTO 1270
1160  PRINT "  YOU OVERSHOT BY ";ABS(D-R[M,C]);" FEET·"
1170  GOTO 1270
1180  N1=N1-1
1190  IF N1>1 THEN 1270
1200  FOR M1=1 TO N
1210  IF P[M1]=12 THEN 1250
1220  PRINT
1230  PRINT "GAME OVER·  ";M1;"WINS·"
1240  STOP
1250  NEXT M1
1260  STOP
1270  PRINT ""
1280  NEXT M
1290  T=T+1
1300  GOTO 650
1310  PRINT
1320  PRINT "THIS IS A WAR GAME·  TWO OR THREE PLAYERS ARE GIVEN"
1330  PRINT "(THEORETICAL) CANNONS WITH WHICH THEY ATTEMPT TO SHOOT EACH"
1340  PRINT "OTHER·  THE PARAMETERS FOR DISTANCES AND MUZZLE VELOCITIES ARE"
1350  PRINT "SET AT THE BEGINING OF THE GAME·  THE SHOTS ARE FIRED BY"
1360  PRINT "GIVING A FIRING ANGLE, EXPRESSED IN DEGREES FROM HORIZONTAL"
1370  PRINT
1380  PRINT "THE COMPUTER WILL KEEP TRACK OF THE GAME AND REPORT ALL"
1390  PRINT "MOVES·  A 'HIT' IS SCORED BY FIRING A SHOT WITHIN 5% OF THE TOTAL"
1400  PRINT "DISTANCE FIRED OVER·  GOOD LUCK"
1410  PRINT ""
1420  GOTO 160
1430  END
```

Instructions here — (not repeated in sample run)

SAMPLE RUN

```
WAR3

THIS IS THE BASIC VERSION OF 'WAR3'·  TWO OR THREE MAY PLAY
DO YOU NEED INSTRUCTIONS?N


NO· OF PLAYERS?3


DISTANCE (FT·)  1     TO  2     ?1000
DISTANCE (FT·)  2     TO  3     ?2000
DISTANCE (FT·)N 3     TO  1     ?2500


MUZZLE VELOCITY (FT·/SEC·) OF  1     ?300
MUZZLE VELOCITY (FT·/SEC·) OF  2     ?350
MUZZLE VELOCITY (FT·/SEC·) OF  3     ?400
```

It's easier if you draw a diagram



```
ROUND  1

PLAYER  1     SHOOTING AT?3
FIRING ANGLE?78
  YOU OVERSHOT BY  144·095     FEET·


PLAYER  2     SHOOTING AT?3
FIRING ANGLE?79
  YOU UNDERSHOT BY  566·111     FEET·


PLAYER  3     SHOOTING AT?1
FIRING ANGLE?76
  YOU UNDERSHOT BY  154·143     FEET·



ROUND  2

PLAYER  1     SHOOTING AT?2
                              FIRING ANGLE?80
  YOU UNDERSHOT BY  1539·64     FEET·


PLAYER  2     SHOOTING AT?3
FIRING ANGLE?76
  YOU UNDERSHOT BY  202·396     FEET·


PLAYER F 3    SHOOTING AT?1
FIRING ANGLE?72
  YOU OVERSHOT BY  149·81     FEET·
```

```
ROUND  3

PLAYER  1     SHOOTING AT?3
                              FIRING ANGLE?80
  YOU UNDERSHOT BY  1536·77     FEET·


PLAYER  2     SHOOTING AT?3
FIRING ANGLE?77
                   A HIT -  3     IS DEFUNCT·
```

Player 2 wiped out Player 3, however, Player 1 is zeroing in on Player 2

```
ROUND  4

PLAYER  1 ·   SHOOTING AT?2
FIRING ANGLE?78
  YOU OVERSHOT BY  144·814     FEET·


PLAYER  2     SHOOTING AT?1
                              FIRING ANGLE?79
  YOU OVERSHOT BY  434·333     FEET·



ROUND  5

PLAYER  1     SHOOTING AT?2
FIRING ANGLE?79
  YOU OVERSHOT BY  53·9304     FEET·


PLAYER  2     SHOOTING AT?1
FIRING ANGLE?8
  YOU UNDERSHOT BY  74·1608     FEET·



ROUND  6

PLAYER  1     SHOOTING AT?2
FIRING ANGLE?8P
  A HIT -  2     IS DEFUNCT·

GAME OVER·   1    WINS·

DONE
```

DR. Z

LISTING

```
LIST
DR.Z

10  DIM A$[72],B$[72]
15  REM  DEVELOPED BY  DR.Z  1972
20  PRINT "HELLO THERE, I'M YOUR COMPUTERTHERAPIST."
30  PRINT "WHAT IS YOUR NAME? AND TELL ME SOMETHING ABOUT YOURSELF."
40  PRINT "HOWEVER, DON'T TYPE MORE THAN ONE LINE. I TIRE EASILY."
50  INPUT A$
60  PRINT "WHAT DID YOU SAY YOUR NAME WAS AGAIN?"
70  INPUT B$
80  PRINT "HOW DO YOU FEEL TODAY?"
90  LET C=U=V=0
100  INPUT A$
110  PRINT
120  PRINT
130  IF C=10 THEN 720
140  LET Z=INT(10*RND(0))
150  IF U=Z THEN 140
160  IF V=Z THEN 140
170  LET U=Z
180  IF Z <> 0 THEN 200
190  GOTO 690
200  IF Z <> 1 THEN 220
210  GOTO 420
220  IF Z <> 2 THEN 240
230  GOTO 450
240  IF Z <> 3 THEN 260
250  GOTO 480
260  IF Z <> 4 THEN 280
270  GOTO 510
280  IF Z <> 5 THEN 300
290  GOTO 540
300  IF Z <> 6 THEN 320
310  GOTO 570
320  IF Z <> 7 THEN 340
330  GOTO 600
340  IF Z <> 8 THEN 360
350  GOTO 630
360  IF Z <> 9 THEN 380
370  GOTO 660
380  GOTO 690
390  PRINT "THAT'S VERY INTERESTING, TELL ME MORE."
400  PRINT
410  GOTO 690
420  PRINT "HAVE YOU FELT THIS WAY LONG?"
430  PRINT
440  GOTO 690
450  PRINT "DO YOU THINK THIS IS REASONABLE IN LIGHT OF YOUR INTERESTS?"
460  PRINT
470  GOTO 690
480  PRINT "DO YOUR FRIENDS FIND THIS ACCEPTABLE?"
490  PRINT
500  GOTO 690
510  PRINT "DO YOU FEEL COMFORTABLE WITH THIS FEELING?"
520  PRINT
530  GOTO 690
540  PRINT "DO YOU THINK THAT THIS IS A NORMAL FEELING?"
550  PRINT
560  GOTO 690
570  PRINT "WHY DO YOU THINK YOU FEEL THIS WAY?"
580  PRINT
590  GOTO 690
600  PRINT "HAVE YOU TALKED TO ANYONE ABOUT THIS?"
610  PRINT
620  GOTO 690
630  PRINT "WHY ARE YOU HERE?"
640  PRINT
650  GOTO 690
660  PRINT "ARE YOU SATISFIED WITH THE WAY YOUR IDEAS ARE DEVELOPING?"
670  PRINT
680  GOTO 690
690  LET C=C+1
700  LET U=Z
710  GOTO 100
720  PRINT "I THINK YOU ARE MAKING A GREAT ATTEMPT TO SOLVE YOUR"
730  PRINT "DIFFICULTIES, AND I SEE NO NEED TO CONTINUE THIS"
740  PRINT "SESSION ANY FURTHER."
750  PRINT B$" WOULD YOU MAKE ANOTHER APPOINTMENT WITH MY COMPUTER"
760  PRINT "FOR SOMETIME IN THE NEXT FEW WEEKS.WHAT DATE WOULD YOU"
770  PRINT "PREFER?"
780  INPUT A$
790  PRINT "THAT WILL BE FINE."
800  PRINT "I'VE ENJOYED COMMUNICATING WITH YOU."
810  PRINT "HAVE A NICE DAY."
820  FOR T=1 TO 6
830  PRINT
840  NEXT T
850  END
```

If your BASIC has ON-GOTO statements, you should be able to improve this coding dramatically.

SAMPLE RUN

```
RUN
DR.Z

HELLO THERE, I'M YOUR COMPUTERTHERAPIST.
WHAT IS YOUR NAME? AND TELL ME SOMETHING ABOUT YOURSELF.
HOWEVER, DON'T TYPE MORE THAN ONE LINE. I TIRE EASILY.
?HOMEEE, AND I HUHFGVB DGVXFCSDZX BHVBN JMNHB FEELINGS.
WHAT DID YOU SAY YOUR NAME WAS AGAIN?
?HOMEREE
HOW DO YOU FEEL TODAY?
?LIKE I SAID BEFORE, YOU KNOW.


WHY ARE YOU HERE?

?I FEEL YOU KNOW, UNDERSTAND?


ARE YOU SATISFIED WITH THE WAY YOUR IDEAS ARE DEVELOPING?

?YOU KNOW, HE, HIM, THEM, YOU KNOW.


HAVE YOU FELT THIS WAY LONG?

?I DON'T, YOU KNOW, THAT THIS THING IS, YOU KNOW.


WHY ARE YOU HERE?

?YOU KNOW, IT IS THE THING NOW, YOU KNOW.


HAVE YOU FELT THIS WAY LONG?

?YA, UGH, YOU KNOW.


DO YOU THINK THIS IS REASONABLE IN LIGHT OF YOUR INTERESTS?

?YA, YOU KNOW, THIS IS COMMUNICATION,YOU KNOW.


WHY ARE YOU HERE?

?YOU KNOW, YOU KNOW.


DO YOUR FRIENDS FIND THIS ACCEPTABLE?

?YA, WE ALL COMMUNICATE, YOU KNOW.


HAVE YOU TALKED TO ANYONE ABOUT THIS?

?ALL MY PEOPLE, YOU KNOW.


DO YOU THINK THAT THIS IS A NORMAL FEELING?

?THIS IS IT, YOU KNOW.


I THINK YOU ARE MAKING A GREAT ATTEMPT TO SOLVE YOUR
DIFFICULTIES, AND I SEE NO NEED TO CONTINUE THIS
SESSION ANY FURTHER.
HOMEREE WOULD YOU MAKE ANOTHER APPOINTMENT WITH MY COMPUTER
FOR SOMETIME IN THE NEXT FEW WEEKS.WHAT DATE WOULD YOU
PREFER?
?HOW ABOUT, YOU KNOW.
THAT WILL BE FINE.
I'VE ENJOYED COMMUNICATING WITH YOU.
HAVE A NICE DAY.
```

```
DONE
```

Concentration

by Paul Calter
Vermont Technical College

How good are your personal memory banks? Find out by playing this computer version of the well-known card game, CONCENTRATION.

The entire deck of 52 cards is shuffled, and dealt face down on the table. The player turns over any two cards he pleases. If they are a *pair,* they are removed from play, the player scores one point, and picks again. If they are not a pair, the player tries to memorize the cards and their location for future picks, and returns them face down on the table in the same location.

In this computer version, submitted by James Vanderbeek of Vermont Technical College, the computer "deals" the cards in positions one to fifty-two. There is only one player (the computer does not pick cards in this version) who enters the numbers of the positions chosen. The computer then tells you which cards you have picked. If they are a pair, it removes them from play, and your score is incremented: if not, the identity of the cards is overprinted, and you have a chance to pick again.

After a total of 26 picks, the play ends and your score is printed, along with the computer's candid assessment of your ability.

Some possible expansions of this program are:

1. Play against the computer — where the computer picks its pairs completely at random, from the cards remaining on the "table."
2. Play against the computer — where the computer is blessed with a perfect memory of all cards previously overturned.
3. Play against the computer — where the computer usually remembers the cards previously shown, but sometimes has "lapses" of memory. (How would you program that?) This version would probably provide the most interesting play.
4. Play against another person, using the same terminal.

PROGRAM LISTING

```
190 RANDOMIZE
200 DIM A$(52),C$(52)
210 FOR X = 1 TO 52                               'MAKE THE DECK
220    READ E$
230    LET A$(X) = E$
240 NEXT X
250 ' SHUFFLE AND DEAL
260 FOR Z = 1 TO 1000
270    LET B$ = A$(INT(RND*52+1))
280    FOR V = 0 TO 51
290       IF B$ = C$(V) THEN 330
300    NEXT V
310    LET C$(D) = B$
320    LET D = D+1
330 NEXT Z
340 ' START TO PLAY
350 FOR N = 1 TO 26
360    PRINT "FIRST CARD";
370    INPUT U
380    LET G = 1
390    IF C$(U) = " " THEN 840
400    PRINT "SECOND CARD";
410    INPUT W
420    LET G = 2
430    IF C$(W) = " " THEN 840
440    IF U<>W THEN 470
450    PRINT "YOU CAN'T PICK THE SAME CARD TWICE!"
460    GOTO 400
470    IF SEG$(C$(U),1,1) = SEG$(C$(W),1,1) THEN 580
480    PRINT "#";U;"IS ";C$(U),"#";W;"IS ";C$(W),   'REVEAL THE CARDS
490    FOR I = 1 TO 50
500       PRINT "";
510    NEXT I
520    FOR P = 1 TO 4                      'HIDE THE CARDS
530       PRINT CHR$(13);"@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@";
540    NEXT P
550    PRINT
560    PRINT
570    GOTO 630
580    PRINT "THAT'S A MATCH --";C$(U),C$(W)
590    LET C$(U) =" "                      'TAKE THE PAIR
600    LET C$(W) =" "
610    LET S = S+1
620    PRINT "YOUR SCORE IS NOW";S;" YOU HAVE HAD";N;"PICKS."
630 NEXT N
640 ' THE RESULTS
650 LET S1 = S/(N/4)
660 PRINT "YOU SCORED";S;"OUT OF";N;"THAT IS ";
670 ON S1 + 1.5 GOTO 680,700,720,760
680 PRINT "POOR."
690 GOTO 770
700 PRINT "FAIR."
710 GOTO 770
720 PRINT "GOOD."
730 GOTO 770
740 PRINT "FANTASTIC! ! ! !"
750 GOTO 770
760 PRINT ". . . AAAH. . .UH....YOU MUST HAVE CHEATED!"
770 PRINT
780 PRINT "DO YOU WANT TO TRY AGAIN";
790 INPUT Z$
800 IF F$ = "YES" THEN 260
810 PRINT
820 PRINT "COME BACK AGAIN!!"
830 STOP
840 PRINT "YOU HAVE ALREADY MATCHED THAT CARD."
850 IF G = 1 THEN 360
860 GOTO 400
870 DATA"AS","2S","3S","4S","5S","6S","7S","8S","9S","10S","JS","KS","QS
880 DATA"AH","2H","3H","4H","5H","6H","7H","8H","9H","10H","JH","QH","KH
890 DATA"AD","2D","3D","4D","5D","6D","7D","8D","9D","10D","JD","QD","KD
900 DATA"AC","2C","3C","4C","5C","6C","7C","8C","9C","10C","JC","QC","KC
910 END
```

SAMPLE RUN

```
FIRST CARD? 12
SECOND CARD? 3
●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●

FIRST CARD? 14
SECOND CARD? 51
●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●

FIRST CARD? 12
SECOND CARD? 14
THAT'S A MATCH --JD             JH
YOUR SCORE IS NOW 1  YOU HAVE HAD 3 PICKS.
FIRST CARD? 22
SECOND CARD? 45
●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●

FIRST CARD? 31
SECOND CARD? 36
●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●

FIRST CARD? 45
SECOND CARD? 31
THAT'S A MATCH --QD             QC
YOUR SCORE IS NOW 2  YOU HAVE HAD 6 PICKS.
FIRST CARD? 44
SECOND CARD? 16
●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●

FIRST CARD? 29
SECOND CARD? 50
●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●

FIRST CARD? 29
SECOND CARD? 44
THAT'S A MATCH --10S            10C
YOUR SCORE IS NOW 3  YOU HAVE HAD 9 PICKS.
FIRST CARD? 50
SECOND CARD? 3
●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●

FIRST CARD? 30
SECOND CARD? 6
●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●
```

Another new game from Creative Computing . . . .

ROADRACE

Author: Unknown
Modified by: Bill Cotter, Pittsfield, Mass.

Description: You are the driver of a race car on the notorious NY Route 20. You'll have to drive 5 miles with ½ gallon of gas, while keeping alert for changes in the road conditions, other cars, etc.

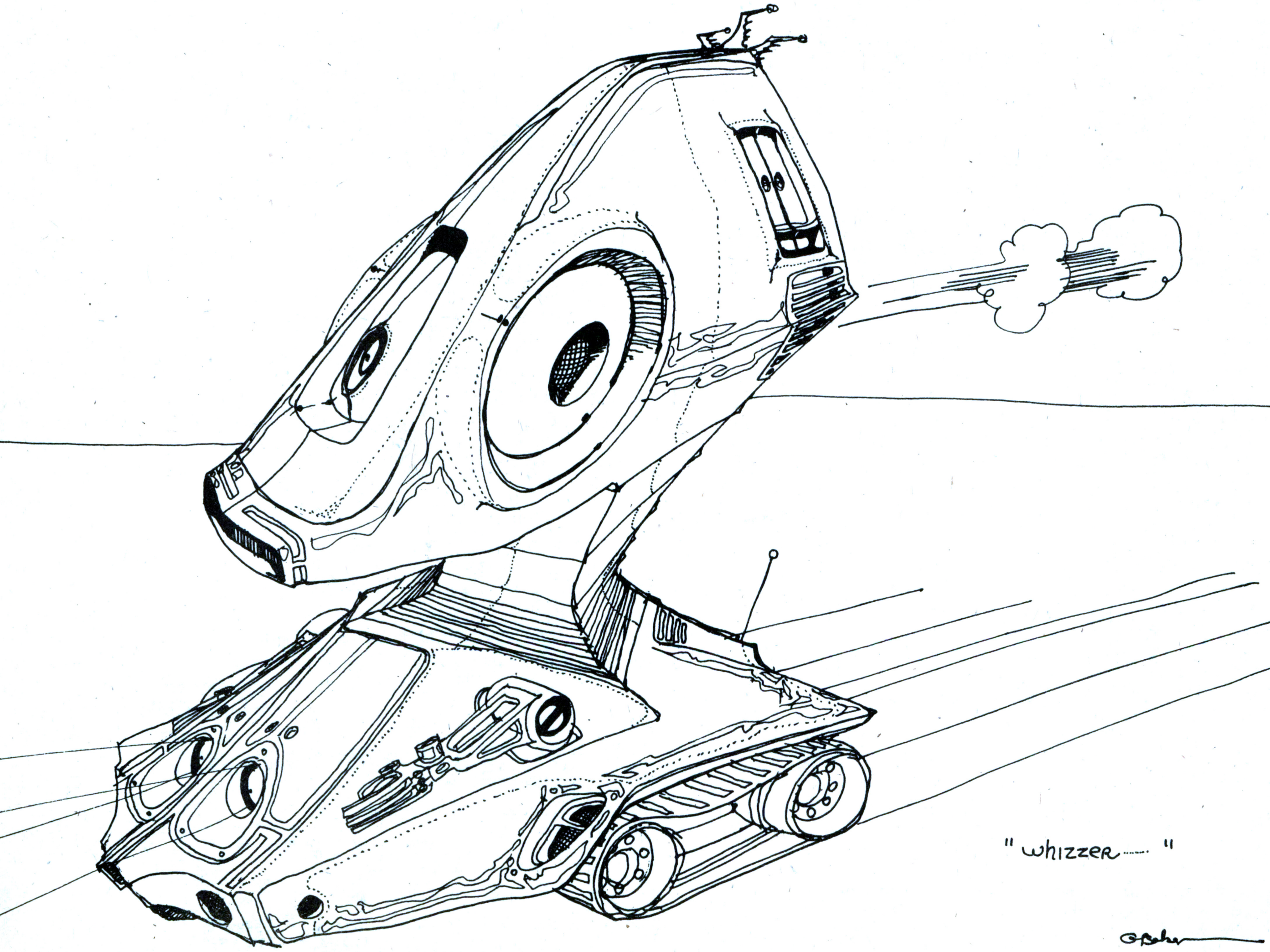
At the start you pick your car and course. During the race you control braking and acceleration.

Watch out for passing another car! If you try to go the same speed he's going, you're going to meet a Greyhound bus head-on!

Suggestions: The game is tough to win. I usually wipe out in a curve or run out of gas. You might want to increase your MPG rating . . . look at line 870.

Good luck!

Listing and run on next page.



ROADRACE

LISTING

```
100  PRINT "           THIS IS THE PITTSFIELD-ALBANY"
110  PRINT "                    ROAD RALLY"
120  PRINT
130 PRINT "WELCOME TO THE FIRST ANNUAL PITTSFIELD-TO-ALBANY ROAD RALLY."
140 PRINT "YOU'LL BE DRIVING DOWN RT. 20, TRYING TO WIN THE RACE AND"
150 PRINT "STAY ALIVE IN THE BARGAIN.  GOOD LUCK!!"
160 PRINT
170 PRINT "YOU HAVE YOUR CHOICE OF: (1) A VW; (2) 283 NOVA;"
180 PRINT "(3) Z-28; OR (4) FERRARI"
190 PRINT
200 PRINT "CHOOSE THE CAR YOU WANT BY THE NUMBER IN FRONT OF IT."
210 PRINT "REMEMBER, THE BETTER THE CAR THE MORE GAS IT USES."
220 PRINT "WHICH CAR";
230  INPUT C1
240  LET C1=INT(C1)
250 IF C1>4 THEN 280
260 IF C1 <1 THEN 280
270 GO TO 300
280 PRINT "INVALID CAR NUMBER. NEW CAR ";
290 GO TO 230
300 PRINT
310 IF N2=1 THEN 350
320 PRINT "YOU NOW CHOOSE WHICH COURSE YOU WANT TO RACE ON."
330 PRINT "THE EASIEST COURSE IS NUMBER 1, AND IS THE STRAIGHTEST"
340 PRINT "ROUTE.  NUMBER 5 CONSISTS MOSTLY OF TURNS AND TWISTS."
350 PRINT "WHICH COURSE DO YOU WANT (1 TO 5)";
360 INPUT C2
370 LET C2=INT(C2)
380 IF C2 < 1 THEN 410
390 IF C2 >5 THEN 410
400 GO TO 430
410 PRINT "INVALID COURSE NUMBER. NEW CHOICE ";
420  GO TO 360
430 IF N2=1 THEN 490
440 PRINT "YOU WILL NEED TO TRAVEL 5 MILES WITH .5 GALLONS OF GAS"
450 PRINT "YOUR STATUS WILL BE SHOWN EACH 10 SECONDS.  AFTER EACH STATUS"
460 PRINT "CHECK YOU WILL BE ASKED FOR A NEW RATE OF GAS.  A RATE OF"
470 PRINT "+10 IS HARD ACCELERATION, AND -10 IS HARD BRAKING.  ANY NUMBER"
480 PRINT "IN BETWEEN IS ALLOWABLE."
490 FOR I=1 TO C1
500 READ B,M,S
510 LET B=B/10
520 NEXT I
530 LET A1=.5
540 LET M1=0
550 LET C1=C1/2
560 LET V=0
570 PRINT
580 LET R1=0
590 LET T=0
600 LET D=0
610 LET Q1=0
620 PRINT "PRESENT VELOCITY = ";V;" NO. OF GALLONS = ";A1
630 PRINT "NO. OF MILES = ";M1;" TIME PASSED = ";T;" SECONDS"
640 IF M1>=5 THEN 1460
650 PRINT "WHAT IS YOUR NEW RATE OF GAS ";
660 INPUT G
670IF G<-10 THEN 700
680 IF G>10 THEN 700
690 GO TO 720
700 PRINT "NOT VALID. NEW RATE ";
710 GO TO 660
720 IF G<9 THEN 780
730 LET Z=Z+1
740 IF Z>4 THEN 760
750 GO TO 790
760 PRINT "YOUR ENGINE BLEW.  YOU GOT HIT BY A PISTON."
770 GO TO 1270
780 LET Z=0
790 LET V=INT(B*G-M*V+V)
800 LET T=T+10
810 PRINT
820 PRINT "ROAD CONDITIONS :";
830 IF V>0 THEN 850
840 LET V=0
850 LET M1=M1+V/460
860 IF G<0 THEN 890
870 LET A1=A1-(G*S)/5000
880 IF A1<0 THEN 1380
890 IF R1=1 THEN 1050
900 IF Q1=1 THEN 980
910  LET Q=INT((C2+1)*RND(X))
920 LET R=INT((3.75-C2)*RND(X))
930 IF R>0 THEN 1290
940 IF Q>0 THEN 1340
950 PRINT "CLEAR AND STRAIGHT"
960 PRINT
970 GO TO 620
980 LET H=INT(15+35.*RND(X))
990 LET H=H+5*C1
1000 IF V>H THEN 1500
1010 PRINT "THROUGH CURVE"
.1020 PRINT
1030 LET Q1=0
1040 GO TO 620
1050 LET E=E-(V-D)*3.0
1060 IF E<0 THEN 1100
1070 PRINT "VEHICLE ";E;" FEET AHEAD"
1080 PRINV
1090 GO TO 620
1100 IF V-D<5 THEN 1180
1110 PRINT "VEHICLE PASSED BY ";
1120 LET D=V-D
1130 PRINT D;
1140 PRINT " MPH"
1150 PRINT
1160 LET R1=0
1170 GO TO 620
1180 PRINT "VEHICLE BEING PASSED "
1190 LET  D=INT(25+40*RND(X))
1200 PRINT "GRAYHOUND BUS IN OTHER LANE ";
1210 PRINT "DOING ";
1220 PRINT D;
1230 PRINT " MPH ";
1240 LET D=V+D
1250 PRINT "CRASH VELOCITY = ";
1260 PRINT D
1270 PRINV "WHERE IS YOUR FUNERAL BEING HELD ?"
1280 GO TO 1560
1290 PRINT "VEHICLE AHEAD 1000 FEET"
1300 PRINT
1310 LET D=INT(25+35*RND(X))
1320 LET R1=1
1330 GO TO 620
1340 PRINT " WARNING: CURVE AHEAD "
1350 LET Q1=1
1360 PRINT
1370 GO TO 620
1380 PRINT "EXCELLENT BUT WAIT!"
1390 PRINT
1400 PRINT "YOU RAN OUT OF GAS"
1410 GO TO 1550
1420 PRINT "BUT SOME HOW YOU MADE IT"
1430 PRINT
1440 LET R1=0
1450 GO TO 620
1460 PRINT
1470 PRINT
1480 PRINT "YOU MADE IT (LUCKY) !!!!!!!"
1490 GO TO 1560
1500  PRINT "ARE TERRIBLE"
1510 LET H=H-5*C1
1520 PRINT H;" WAS THE SPEED THROUGH THE CURVE"
1530 PRINT V;" WAS YOUR SPEED. BY THE WAY ";
1540 GO TO 1270
1550 PRINT "YOU LEAD FOOTED *#$%#$#&"
1560 PRINT "YOU WANT TO TRY AGAIN, RIGHT !!!!"
1570 PRINT "1-YES, 2-NO ";
1580 INPUT V
1590 IF V=2 THEN 1620
1600 N2=1
1610 GO TO 1640
1620 PRINT "CHICKEN"
1630 GO TO 1700
1640 RESTORE
1650 GO TO 220
1660 DATA 45,.53,10
1670 DATA 60,.5,13
1680 DATA 70,.41,15
1690 DATA 80,.39,18
1700 END
```

SAMPLE RUN →

```
WELCOME TO THE FIRST ANNUAL PITTSFIELD-TO-ALBANY ROAD RALLY.
YOU'LL BE DRIVING DOWN RT. 20, TRYING TO WIN THE RACE AND
STAY ALIVE IN THE BARGAIN.  GOOD LUCK!!

YOU HAVE YOUR CHOICE OF: (1) A VW; (2) 283 NOVA;
(3) Z-28; OR (4) FERRARI

CHOOSE THE CAR YOU WANT BY THE NUMBER IN FRONT OF IT.
REMEMBER, THE BETTER THE CAR THE MORE GAS IT USES.
WHICH CAR ?1

YOU NOW CHOOSE WHICH COURSE YOU WANT TO RACE ON.
THE EASIEST COURSE IS NUMBER 1, AND IS THE STRAIGHTEST
ROUTE.  NUMBER 5 CONSISTS MOSTLY OF TURNS AND TWISTS.
WHICH COURSE DO YOU WANT (1 TO 5) ?2
YOU WILL NEED TO TRAVEL 5 MILES WITH .5 GALLONS OF GAS
YOUR STATUS WILL BE SHOWN EACH 10 SECONDS.  AFTER EACH STATUS
CHECK YOU WILL BE ASKED FOR A NEW RATE OF GAS.  A RATE OF
+10 IS HARD ACCELERATION, AND -10 IS HARD BRAKING.  ANY NUMBER
IN BETWEEN IS ALLOWABLE.

PRESENT VELOCITY =     0  NO. OF GALLONS =    .5
NO. OF MILES =      0  TIME PASSED =      0  SECONDS
WHAT IS YOUR NEW RATE OF GAS  ?10

ROAD CONDITIONS : WARNING: CURVE AHEAD

PRESENT VELOCITY =    45  NO. OF GALLONS =   .48
NO. OF MILES =    .0978261  TIME PASSED =    10  SECONDS
WHAT IS YOUR NEW RATE OF GAS  ?2
```

← SLOW DOWN FOR CURVE

```
ROAD CONDITIONS :THROUGH CURVE

PRESENT VELOCITY =    30  NO. OF GALLONS =  .476
NO. OF MILES =    .1630435  TIME PASSED =    20  SECONDS
WHAT IS YOUR NEW RATE OF GAS  ?8

ROAD CONDITIONS :CLEAR AND STRAIGHT

PRESENT VELOCITY =    50  NO. OF GALLONS =   .46
NO. OF MILES =    .2717391  TIME PASSED =    30  SECONDS
```

HARD ACCELERATION →

```
WHAT IS YOUR NEW RATE OF GAS  ?10

ROAD CONDITIONS :VEHICLE AHEAD 1000 FEET

PRESENT VELOCITY =    68  NO. OF GALLONS =   .44
NO. OF MILES =    .4195652  TIME PASSED =    40  SECONDS
WHAT IS YOUR NEW RATE OF GAS  ?4

ROAD CONDITIONS :VEHICLE PASSED BY     6  MPH

PRESENT VELOCITY =    49  NO. OF GALLONS =  .432
NO. OF MILES =    .5260869  TIME PASSED =    50  SECONDS
WHAT IS YOUR NEW RATE OF GAS  ?10

ROAD CONDITIONS :VEHICLE AHEAD 1000 FEET

PRESENT VELOCITY =    68  NO. OF GALLONS =  .412
NO. OF MILES =    .673913  TIME PASSED =    60  SECONDS
WHAT IS YOUR NEW RATE OF GAS  ?5

ROAD CONDITIONS :VEHICLE PASSED BY    22  MPH

PRESENT VELOCITY =    54  NO. OF GALLONS =  .402
NO. OF MILES =    .7913043  TIME PASSED =    70  SECONDS
WHAT IS YOUR NEW RATE OF GAS  ?10

ROAD CONDITIONS : WARNING: CURVE AHEAD

PRESENT VELOCITY =    70  NO. OF GALLONS =  .382
NO. OF MILES =    .9434782  TIME PASSED =    80  SECONDS
WHAT IS YOUR NEW RATE OF GAS  ?4
```

← DIDN'T SLOW ENOUGH FOR CURVE

```
ROAD CONDITIONS :ARE TERRIBLE
   39  WAS THE SPEED THROUGH THE CURVE
   50  WAS YOUR SPEED. BY THE WAY WHERE IS YOUR FUNERAL BEING HELD ?
```

Another new game from Creative Computing....

Condot*

as reviewed by
Peter Olivieri
Boston College

You don't have to have connections to connect with this program. It's the old childhood favorite of "connect the dots." The objective, however, is not to draw a picture by connecting the dots, but rather to carve out squares of "real estate" with the computer as an able adversary. The player who connects the two dots which complete a square gets ownership of that square. In addition, the player gets the added bonus of moving once more. This can be quite advantageous in certain situations. A nine-square grid is provided as a playing board.

The instructions (see REM statements in program) are not as clear as they might be. When you wish to connect two dots you must type in the coordinates (row, column) of the empty space between the dots. However, in identifying what the coordinates are, you must count the "dots" of the grid also. Lest this be equally confusing, a sample of the grid follows complete with an identification of each coordinate where a line may be drawn.

Grid is on page with listing.

In games that I played, the same moves could be replicated in a succeeding game. Thus, once you discover a winning game, you cannot lose (this may be a function of the particular random number generator in use). You'll find it interesting to note that the computer mirrors the player's move in so far as possible. You may also find that the game moves rather slowly, especially for the first three or four moves. Be patient! Once squares begin to fall, the games moves swiftly to its conclusion.

There are some modifications that you may wish to consider if you are going to adopt this program for regular use. In addition to improving the REM statements in the program, I would suggest:

1. Modifying the program so that the grid is printed after *both* players have moved (rather than each time a move is made);
2. Modifying the program so that once a player had ownership of a majority of the squares, the game would end rather than proceed to its inevitable conclusion; and
3. Modify the program so that the player's initials appear in each square he captures.

An interesting problem developed in the running of this program that you should be aware of. One particular sequence of moves resulted in a "hangup" in the execution of the program. This may be the result of a programming error, a typing error somewhere along the line, a system problem, or whatever. A few of us spent a little time debugging, but to no avail. In any case, five runs using the following sequence of moves resulted in an average execution time (until interrupt) of 40 CPU seconds.

Your moves:	1.	1,2
	2.	2,1
	3.	2,3
	4.	5,6
	5.	2,5
	6.	4,5
	7.	6,1
	8.	7,2

A sample run of a game (in this instance, the player being triumphant) follows. While the game may quickly lose its interest to all but those with nothing else to do, the program itself is well worth examining. Creating the flowcharts and source statements for this game would be a worthwhile endeavor, particularly as a project in a programming course. Furthermore, even the process of tracing through the existing source program to discover the game's algorithm would be a worthwhile exercise for the knowledgeable programmer.

Listing and run of CONDOT 1 page over.

---

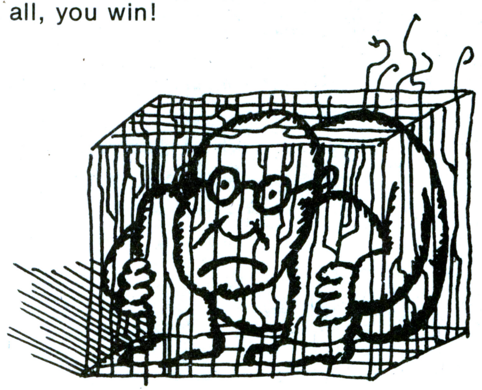
*CONDOT was written by Chuck Lund, St. Paul Public Schools, MN.

Another new game from Creative Computing....

CHASE

Author: Unknown
Modified by: Bill Cotter, Pittsfield, Mass.
Language: BASIC (Honeywell 600/6000)

Description: CHASE puts you in a maze made up of high-voltage fences and posts. This in itself isn't too unpleasant but there're also the five interceptor robots bent on just one thing—your destruction. If these robots touch you . . . that's the end of the game (and you!). There's one hope—make the robots hit the maze, or each other (they're like people—sometimes they'd rather be alone). If you destroy them all, you win!



Listing and run of CHASE on the next page.

CHASE

LISTING

```
100 REM CHASE
110 REM AUTHOR: UNKNOWN
120 REM MODIFIED TO RUN ON HONEYWELL 600/6000 BY BILL COTTER
130 PRINT"YOU ARE WITHIN THE WALLS OF A HIGH VOLTAGE MAZE"
140 PRINT"THERE ARE FIVE SECURITY MACHINES TRYING TO DESTROY YOU"
150 PRINT"YOU ARE THE '*'      THE INTERCEPTORS ARE THE '+'"
160 PRINT"THE AREAS MARKED 'X' ARE HIGH VOLTAGE"
170 PRINT"YOUR ONLY CHANCE FOR SURVIVAL IS TO MANUVER EACH "
180 PRINT"INTERCEPTOR INTO AN 'X'. ---- GOOD LUCK ----"
190 PRINT"MOVES ARE:  1.2.3"\PRINT"            8.*.4"
200 PRINT"            7.6.5"\PRINT
210 DIM A(10,20),N(2)
220 FOR B=1 TO 10
230 FOR C=1 TO 20
240 LET X=INT(10*RND(-1))
250 IF X=5 THEN 280
260 LET A(B,C)=ASC( )
270 GOTO 290
280 LET A(B,C)=ASC(X)
290 NEXT C
300 NEXT B
310 FOR D=1 TO 10
320 LET A(D,1)=ASC(X)
330 NEXT D
340 FOR E=1 TO 10
350 LET A(E,20)=ASC(X)
360 NEXT E
370 FOR F=1 TO 20
380 LET A(1,F)=ASC(X)
390 NEXT F
400 FOR G=1 TO 20
410 LET A(10,G)=ASC(X)
420 NEXT G
430 GOTO 500
440 LET H=INT(1+(10*RND(-1)))\IF H>10 THEN 440
450 LET I=INT(1+(20*RND(-1)))\IF I>20 THEN 450
460 IF A(H,I)=ASC(X) THEN 440
470 IF A(H,I)=ASC(*) THEN 440
480 IF A(H,I)=ASC(+) THEN 440
490 RETURN
500 GOSUB 440
510 LET A(H,I)=ASC(*)\LET J=H\LET K=I
520 GOSUB 440
530 LET A(H,I)=ASC(+)\LET L=H\LETM=I
540 GOSUB 440
550 LET A(H,I)=ASC(+)\LET N=H\LET O=I
560 GOSUB 440
570 LET A(H,I)=ASC(+)\LET P=H\LET Q=I
580 GOSUB 440
590 LET A(H,I)=ASC(+)\LET R=H\LET S=I
600 GOSUB 440
610 LET A(H,I)=ASC(+)\LET T=H\LETU=I
620 N(0)=1
630 FOR D2=1 TO 10
640 FOR B2=1 TO 20
650 N(1)=A(D2,B2)\CHANGE N TO N$\PRINT N$;
660 NEXT B2
670 PRINT
680 NEXT D2
690 INPUT Y\IF Y=0 THEN 800
700 LET V=J\LET W=K
710 ON Y GOTO 720,730,740,750,760,770,780,790
720 LET J=J-1\LET K=K-1\GOTO 800
730 LET J=J-1\GOTO 800
740 LET J=J-1\LET K=K+1\GOTO 800
750 LET K=K+1\GOTO 800
760 LET J=J+1\LET K=K+1\GOTO 800
770 LET J=J+1\GOTO 800
780 LET J=J+1\LET K=K-1\GOTO 800
790 LET K=K-1
800 IF A(J,K)=ASC(X) THEN 1160
810 LET A(V,W)=ASC( )\LET A(J,K)=ASC(*)
820 GOTO 940
830 IF A(X,Y)=ASC(X) THEN 910
840 LET V=X\LET W=Y
850 LET X=SGN(J-X)\LET Y=SGN(K-Y)
860 LET X=X+V\LET Y=Y+W
870 IF A(X,Y)=ASC(*) THEN 920\IF A(X,Y)=ASC( ) THEN 890
880 LET A(V,W)=ASC( )\RETURN
890 LET A(X,Y)=ASC(+)
900 LET A(V,W)=ASC( )
910 RETURN
920 PRINT "*** YOU HAVE BEEN DESTROYED BY A LUCKY COMPUTER ***"
930 GO TO 1180
940 LET X=L\LET Y=M\GOSUB 830
950 LET L=X\LET M=Y
960 LET X=N\LET Y=O\GOSUB 830
970 LET N=X\LET O=Y
980 LET X=P\LET Y=Q\GOSUB 830
990 LET P=X\LET Q=Y
1000 LET X=R\LET Y=S\GOSUB 830
1010 LET R=X\LET S=Y
1020 LET X=T\LET Y=U\GOSUB 830
1030 LET T=X\LET U=Y
1040 IF A(L,M)=ASC(X) THEN 1060
1050 GOTO 630
1060 IF A(N,O)=ASC(X) THEN 1080
1070 GOTO 630
1080 IF A(P,Q)=ASC(X) THEN 1100
1090 GOTO 630
1100 IF A(R,S)=ASC(X) THEN 1120
1110 GOTO 630
1120 IF A(T,U)=ASC(X) THEN 1140
1130 GOTO 630
1140 PRINT"YOU HAVE DESTROYED ALL YOUR OPPONENTS-THE GAME IS YOURS"
1150 GO TO 1180
1160 PRINT"YOU TOUCHED THE FENCE !!!!!!!!!!!"
1170 PRINT "***** ZAP ***** YOU'RE DEAD!!!"
1180 PRINT "ANOTHER GAME (YES OR NO)";
1190 INPUT N9$
1200 IF N9$<>"YES" THEN 1220
1210 GO TO 210
1220 END

ready

*
```

SAMPLE RUN

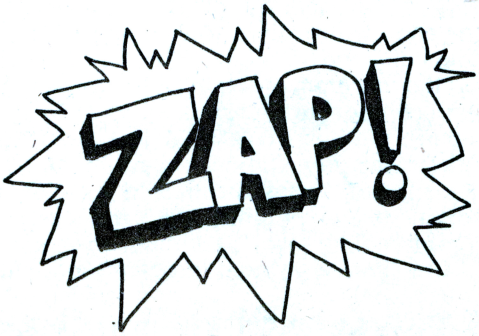
```
YOU ARE WITHIN THE WALLS OF A HIGH VOLTAGE MAZE
THERE ARE FIVE SECURITY MACHINES TRYING TO DESTROY YOU
YOU ARE THE '*'      THE INTERCEPTORS ARE THE '+'
THE AREAS MARKED 'X' ARE HIGH VOLTAGE
YOUR ONLY CHANCE FOR SURVIVAL IS TO MANUVER EACH
INTERCEPTOR INTO AN 'X'. ---- GOOD LUCK ----
MOVES ARE:  1.2.3
            8.*.4
            7.6.5

XXXXXXXXXXXXXXXXXXXX
X X   X      X     X
X   X+       X  X  X
X  +               X
X      X  + +      X
X         *       XX
XX    X           +X
XX   XX  X      X  X
X             X    X
XXXXXXXXXXXXXXXXXXXX
?7
XXXXXXXXXXXXXXXXXXXX
X X   X      X     X
X   X        X  X  X
X     +            X
X   +  X           X
X        + +      XX
XX    X  *       + X
XX   XX  X      X  X
X             X    X
XXXXXXXXXXXXXXXXXXXX
?7
XXXXXXXXXXXXXXXXXXXX
X X   X      X     X
X   X        X  X  X
X                  X
X      X           X
X    +            XX
XX    X + +        X
XX   XX *X      X  X
X             X    X
XXXXXXXXXXXXXXXXXXXX
?5
YOU HAVE DESTROYED ALL YOUR OPPONENTS-THE GAME IS YOURS
```

Why was "7" a good move here?

AND ANOTHER RUN

```
XXXXXXXXXXXXXXXXXXXX
X           XX X   X
X   X X      X     X
X     +            X
X X +X     +   +   X
X           X+     X
X       X   *      X
X          X       X
X                  X
XXXXXXXXXXXXXXXXXXXX
?8
XXXXXXXXXXXXXXXXXXXX
X           XX X   X
X   X X      X     X
X                  X
X X  X +           X
X    +     +X +    X
X       X  *+      X
X          X       X
X                  X
XXXXXXXXXXXXXXXXXXXX
?7
XXXXXXXXXXXXXXXXXXXX
X           XX X   X
X   X X      X     X
X                  X
X X  X             X
X       +   X      X
X     + X +  +     X
X         *X       X
X                  X
XXXXXXXXXXXXXXXXXXXX
?5
XXXXXXXXXXXXXXXXXXXX
X           XX X   X
X   X X      X     X
X                  X
X X  X             X
X           X      X
X       X+         X
X      +   X+      X
X          *       X
XXXXXXXXXXXXXXXXXXXX
?1
*** YOU HAVE BEEN DESTROYED BY A LUCKY COMPUTER ***
```

Would a move of "8" have been better?



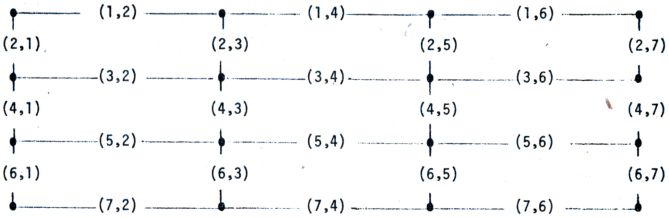
CONDOT LISTING

```
1 PRINT"THIS PROGRAM WILL PLAY CONNECT THE DOTS WITH YOU."
2 PRINT "THE GAME IS PLAYED ON A 4 X 4 ARRAY. WHEN"
3 PRINT"YOU WANT TO MAKE A MOVE YOU MUST TYPE IN"
4 PRINT"THE COORDINATES OF THE SPACE BETWEEN THE TWO DOTS YOU"
5 PRINT"WISH TO CONNECT. ENTER EACH OF YOUR MOVES BY TYPING"
6 PRINT"THE ROW NUMBER, A COMMA AND THEN THE COLUMN NUMBER."
7 PRINT"THE UPPER LEFT HAND CORNER OF THE ARRAY IS 1,3."
8 PRINT"HERE WE GO."
20 DIM A(12,32)
30 V=0
40 FOR R=1 TO 12
50 FOR C=1 TO 12
60 IF R/2=INT(R/2) THEN 100
70 IF C/2=INT(C/2) THEN 100
80 A(R,C)=-50
90 GO TO 110
100 A(R,C)=0
110 NEXT C
120 NEXT R
130 IF V=1 THEN 200
200 GO SUB 1000
210 PRINT"YOUR MOVE";
220 INPUT X,Y
230 IF X=INT(X/1) THEN 260
240 PRINT"TYPING ERROR"
250 GO TO 210
260 IF (X-1)*(X-7)>0 THEN 240
265 X=X+2
270 IF (Y-1)*(Y-7)>0 THEN 240
272 Y=Y+2
280 IF (X+Y+1)/2<>INT((X+Y+1)/2) THEN 240
290 IF A(X,Y)<>0 THEN 240
300 A(X,Y)=50
310 IF X/2=INT(X/2) THEN 380
320 IF A(X-2,Y)+A(X-1,Y+1)+A(X-1,Y-1)<>150 THEN 350
330 P=1
340 A(X-1,Y)=1
350 IF A(X+2,Y)+A(X+1,Y+1)+A(X+1,Y-1)<>150 THEN 440
360 A(X+1,Y)=1
370 GO TO 200
380 IF A(X,Y-2)+A(X+1,Y-1)+A(X-1,Y-1)<>150 THEN 410
390 A(X,Y-1)=1
400 P=1
410 IF A(X,Y+2)+A(X+1,Y+1)+A(X-1,Y+1)<>150 THEN 440
420 A(X,Y+1)=1
430 GO TO 200
440 IF P=1 THEN 200
450 GO SUB 1000
460 PRINT"MY MOVE"
470 FOR R=4 TO 10 STEP 2
480 FOR C=4 TO 10 STEP 2
490 IF A(R-1,C)+A(R+1,C)+A(R,C-1)+A(R,C+1)<>150 THEN 680
500 A(R,C)=-1
510 IF A(R-1,C)<>0 THEN 550
520 A(R-1,C)=50
530 IF A(R-3,C)+A(R-2,C-1)+A(R-2,C+1)<>150 THEN 450
540 LET A(R-2,C)=-1
550 IF A(R+1,C)<>0 THEN 590
560 A(R+1,C)=50
570 IF A(R+3,C)+A(R+2,C-1)+A(R+2,C+1)<>150 THEN 450
580 A(R+2,C)=-1
590 IF A(R,C-1)<>0 THEN 630
600 A(R,C-1)=50
610 IF A(R,C-3)+A(R-1,C-2)+A(R+1,C+2)<>150 THEN 450
620 A(R,C-2)=-1
630 IF A(R,C+1)<>0 THEN 450
640 A(R,C+1)=50
650 IF A(R,C+3)+A(R-1,C+2)+A(R+1,C+2)<>150 THEN 450
660 A(R,C+2)=-1
670 GO TO 450
680 NEXT C
690 NEXT R
692 IF E>1 THEN 730
700 IF A(12-X,32-Y)<>0 THEN 730
710 A(12-X,32-Y)=50
712 IF E>1 THEN 870
720 GO TO 200
730 FOR R=3 TO 9
740 FOR C=3 TO 9
750 IF (R+C)/2=INT((R+C)/2) THEN 850
760 IF A(R,C)<>0 THEN 850
780 IF R/2=INT(R/2) THEN 830
790 IF A(R-2,C)+A(R-1,C-1)+A(R-1,C+1)=100 THEN 850
800 IF A(R+2,C)+A(R+1,C-1)+A(R+1,C+1)=100 THEN 850
810 A(R,C)=50
820 GO TO 200
830 IF A(R,C-2)+A(R-1,C-1)+A(R+1,C-1)=100 THEN 850
840 IF A(R,C+2)+A(R-1,C+2)+A(R+1,C+1)<>100 THEN 810
850 NEXT C
860 NEXT R
862 IF E>1 THEN 700
870 R=INT(RND(R)*7)+3
880 C=INT(RND(C)*7)+3
881 IF R/2=INT(R/2) THEN 885
882 IF C/2=INT(C/2) THEN 900
883 GO TO 870
885 IF C/2<>INT(C/2) THEN 900
886 GO TO 870
900 IF A(R,C)<>0 THEN 870
910 A(R,C)=50
920 GO TO 200
930 PRINT"DO YOU WANT TO PLAY AGAIN(TYPE 1 FOR YES OR 2 FOR NO)";
940 INPUT B
950 IF B=1 THEN 40
960 STOP
1000 P=0
1010 D=0
1020 E=0
1030 FOR R=3 TO 9
1040 FOR C=3 TO 9
1050 IF A(R,C)<>0 THEN 1080
1060 PRINT"   ";
1070 GO TO 1240
1080 IF A(R,C)<>-50 THEN 1110
1090 PRINT" . ";
1100 GO TO 1240
1110 IF A(R,C)<>-1 THEN 1140
1120 PRINT" C ";
1130 GO TO 1170
1140 IF A(R,C)<>1 THEN 1200
1150 PRINT" H ";
1170 D=D+A(R,C)
1180 E=E+1
1190 GO TO 1240
1200 IF R/2=INT(R/2) THEN 1230
1210 PRINT" - ";
1220 GO TO 1240
1230 PRINT " : ";
1240 NEXT C
1245 PRINT
1250 NEXT R
1260 IF E>=9 THEN 1280
1270 RETURN
1280 IF D>0 THEN 1310
1290 PRINT "I WON"
1300 GO TO 930
1310 PRINT 'YOU WON'
1320 GO TO 930
1400 END
```

Can you figure out how to make the grid larger or smaller?

GRID

Here's a sample of the grid showing the Coordinates for each possible move



SAMPLE RUN

```
THIS PROGRAM WILL PLAY CONNECT THE DOTS WITH YOU.
THE GAME IS PLAYED ON A 4 X 4 ARRAY. WHEN
YOU WANT TO MAKE A MOVE YOU MUST TYPE IN
THE COORDINATES OF THE SPACE BETWEEN THE TWO DOTS YOU
WISH TO CONNECT. ENTER EACH OF YOUR MOVES BY TYPING
THE ROW NUMBER, A COMMA AND THEN THE COLUMN NUMBER.
THE UPPER LEFT HAND CORNER OF THE ARRAY IS 1,1.
HERE WE GO.
 .     .     .     .

 .     .     .     .

 .     .     .     .

 .     .     .     .
YOUR MOVEV3,4
 .     .     .     .

 .     .  -  .     .

 .     .     .     .

 .     .     .     .
MY MOVE
 .     .     .     .

 .     .  -  .     .

 .     .  -  .     .

 .     .     .     .
YOUR MOVEV3,2
 .     .     .     .

 .  -  .  -  .     .

 .     .  -  .     .

 .     .     .     .
MY MOVE
 .     .     .     .

 .  -  .  -  .     .

 .     .  -  .  -  .

 .     .     .     .
YOUR MOVEV4,7
```

We didn't show all of the sample run. We skipped from here to here.

```
 .  -  .  -  .  -  .
 :  C  :  C  :
 .  -  .  -  .  -  .
 :  H  :  H  :  H  :
 .  -  .  -  .  -  .
 :  H  :
 .  -  .     .  -  .
YOUR MOVEV2,7
 .  -  .  -  .  -  .
 :  C  :  C  :  H  :
 .  -  .  -  .  -  .
 :  H  :  H  :  H  :
 .  -  .  -  .  -  .
 :  H  :
 .  -  .     .  -  .
YOUR MOVEV6,5
 .  -  .  -  .  -  .
 :  C  :  C  :  H  :
 .  -  .  -  .  -  .
 :  H  :  H  :  H  :
 .  -  .  -  .  -  .
 :  H  :     :
 .  -  .     .  -  .
MY MOVE
 .  -  .  -  .  -  .
 :  C  :  C  :  H  :
 .  -  .  -  .  -  .
 :  H  :  H  :  H  :
 .  -  .  -  .  -  .
 :  H  :  C  :
 .  -  .  -  .  -  .
MY MOVE
 .  -  .  -  .  -  .
 :  C  :  C  :  H  :
 .  -  .  -  .  -  .
 :  H  :  H  :  H  :
 .  -  .  -  .  -  .
 :  H  :  C  :  C  :
 .  -  .  -  .  -  .
YOU WON
```

creative computing
LIBRARY

Four sets of books covering major topics in computer literacy.

COMPUTER GAMES

101 BASIC Computer Games- Dave Ahl
An anthology of games and simulations- from Acey-Deucey to Yahtzee, all in the BASIC language. Contains a complete listing and sample run of each game, plus a descriptive write-up. Large Format. 256 pp. **$7.50**

What To Do After You Hit Return- Bob Albrecht
Another collection of games and simulations- all in BASIC- including number guessing games, word games, hide-and-seek games, pattern games, board games, business and social science simulations and science fiction games. Large format. 158 pp. **$6.95**

Fun And Games With The Computer- Ted Sage
Teaches problem-solving, flow charting and computer programming (in BASIC) in the context of well-known games of chance and strategy. 351 pp. **$5.95**

HAND CALCULATORS

Games, Tricks and Puzzles For A Hand Calculator- Wally Judd
This book is a necessity for anyone who owns or intends to buy a hand calculator, from the most sophisticated (the HP65 for example) to the basic "four banger". 110 pp. **$2.95**

The Calculating Book- Jim Rogers
Discover where you can buy gas during the oil crisis. An anthology of games, puzzles, puns, magic tricks and math problems that can be performed or solved with the pocket calculator. 125 pp. **$2.95**

150 Problems In Crypt-Arithmetic- Maxey Brooke
More grist for the calculator. 156 problems in which letters are substituted for numbers. Examine the number relations between the groups of symbols and solve the problem. 72 pp. **$1.25**

Advanced Applications For Pocket Calculators- Jack Gilbert
This book shows what you can do with your calculator after you balance your checkbook. As much as 80 or 90 percent of the capacity of the calculator is not used, and this book shows you how to use it- step-by-step- regardless of type. Explores all functions of all existing calculators, and all scientific and business applications. 304 pp. **$5.95**

COMPUTERS IN MATH

Problem Solving With The Computer- Ted Sage
Teaches problem solving, chart flowing, and computer programming (in BASIC) in the context of the traditional junior/senior high school curriculum precalculus math. 244 pp. **$4.95**

Problems For Computer Solution- Gruenberger & Jaffray
A collection of 92 problems in engineering, business, social science and mathematics. Oriented toward the FORTRAN language. 401 pp. **$7.25**

My Computer Likes Me- Bob Albrecht
This workbook introduces the BASIC computer language to young or old. The teaching examples are drawn from population problems and demographic data. Large format. 64 pp. **$2.19**

COMPUTERS IN SOCIETY

Electric Media- Les Brown and Sema Marks
How TV and computers affect individuals and society. Cable TV, air waves as public domain, popular and distorted views of the computer speed and storage capacities of computers, communicating with computers, artificial intelligence how computers learn, creative computers. Guest spots by Buzz Aldrin, Dick Cavett and Marshall McLuhan. 168 pp. **$4.95**

Freedom's Edge- Milton Wessel
The computer threat to society. The author, an attorney, shows some of the ways in which the computer is changing our lives- or soon will be. Discusses the data bank, point-of-sale marketing and free competition, computer related crime, controlling the computer, etc. 137 pp. **$4.95**

Computer Lib/Dream Machine- Ted Nelson
This book is devoted to the premise that everybody should understand computers. In a blithe manner the author covers interactive systems, terminals, computer languages, data structures, binary patterns, computer architecture, minicomputers, big computers, microprocessors, simulation, military uses of computers, computer companies, and much, much more. Large format. 127 pp. **$7.00**

Computers in Society- Donald Spencer
How can the computer help the Businessman, artist, or sports announcer? This book examines a wide range of up to date applications of the computer to medicine, engineering, transportation, business, the arts, education, law, process control and many other areas. 208 pp. **$4.95**

Please send me the following books:

Postage and handling	$0.75
Total	

☐ **Payment enclosed**
☐ **Bill me ($1.00 charge)**

THE CREATIVE COMPUTING LIBRARY
42 PLEASANT STREET NEWBURYPORT, MASS. 01950

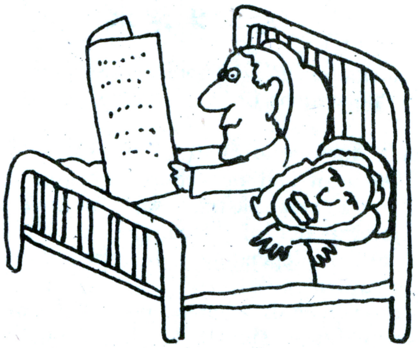
Name ____________________

Street ____________________

City ____________________

State ____________________ Zip ____________

CREATIVE COMPUTING
Reviews



Reviews Editor: Peter Kugel, School of Management, Boston College, Chestnut Hill, MA 02167.

Readers: Want to be a reviewer? Write to the Reviews Editor directly. Publishers: send materials for review to the Reviews Editor.

Fun and Games with the Computer. Edwin R. Sage. 351 pages, $5.95. Entelek, 42 Pleasant Street, Newburyport, MA 01950. 1975.

The author proposes to teach computer programming, using the BASIC language, through games. The reader will have mastered the fundamentals of programming and will have learned one computer language, BASIC, if all suggestions for programming and exercises are worked out.

The book is arranged in 7 chapters, each carefully explaining the rules of the game to be programmed, a flowchart of the procedure for playing the game followed by the BASIC program. The latter half of each chapter deals with improving the program, adding 'personality' and checks to ensure the rules are followed by the computer's opponent. As the games become more complicated, ranging from number guessing to Blackjack, so do the programming and language requirements, thus by the end of the book, the computer user has a good feel for the capabilities of the BASIC language.

Although the author requires an interactive computer system, the book does not require any specific system. Exercises in Chapter 1, and reminders throughout the book, direct the reader to inquire into the local systems requirements for running the BASIC language.

The book, as the author suggests, could be used as a supplementary text for a course involving gaming.

The book is recommended for those involved in teaching the BASIC language or anyone wanting to learn the BASIC language. One ends with a small library of interesting games which can be used for demonstration purposes, if the computer system being used does not have such games present.

John R. Jackobs
Coe College
Cedar Rapids, Iowa

(Available from
Creative Computing Library.)

101 BASIC Computer Games. David Ahl (Ed.), 249 pp., $7.50. Digital Equipment Corp., Maynard, MA. 1974. (Available from *Creative Computing* Library. See ad.)

The paperback book, *101 BASIC Computer Games* contains brief descriptions, BASIC listings and runnings of 101 "games". The programs were collected from a great variety of sources and range from simple picture-printing routines to involved simulation games. As intended, *101 BASIC Computer Games* provides instant stimulation and motivation to intermediate level programmers.

This 8½ by 11 inch book is clearly and conversationally written with legible reproductions of program listings. My junior high school students had no difficulty reading the descriptions preceding the game programs. Interspersed throughout the text are many clever cartoon sketches. Other than the use of the word, "varied", it is impossible to characterize the games included. Mathematical games, simulation games, card games, and sports games are all represented. Titles range from the familiar (NIM, POKER, BASEBL, LIFE) to the surprising (ANIMAL, MUGWMP, SPACWR, ZOOP). The level of the programming is uneven and a variety of versions of BASIC is used. My students delighted in "improving the original" and debugging. Often the latter operation consisted of translating the program into one of our dialects of BASIC (H-P or IBM 370). This translation process itself proved to be an interesting and simulating exercise.

101 BASIC Computer Games is a valuable resource book for teachers and students of BASIC. Its utilization will depend on the course objectives and on your teaching strategy. I have found it to be useful as a source of project ideas for students who have already gained some fluency in BASIC. Additional project ideas and exercises are contained in the companion volume described below.

Patrick Corry
Irvington High School
Irvington, New York

*****

Understanding Mathematics and Logic Using BASIC Computer Games, David Ahl, 60 pp, $1.50. Digital Equipment Corp., Maynard, MA. 1974.

This book extends and analyzes many of strategies and ideas involved in the games contained in *101 BASIC Computer Games.* In this extension, the author discusses logical strategies, gives exercises, and suggests projects related to some of the 101 games. There is a brief but illuminating introduction which includes some thoughts about the use of teams and games as teaching tools. The use of mathematical logic in games is discussed with several examples. The exposition is clear and brief and is intelligible to a high school student. The exercises and project suggestions will be especially useful to those teachers who wish to use *101 BASIC Computer Games* as a source for a computer literacy course. All teachers of BASIC programming should find this volume useful as a source of ideas for discussion and individual projects.

Patrick Corry
Irvington High School
Irvington, New York

*****

What To Do After You Hit Return, 157 pp., $6.95. People's Computer Co./Hewlett-Packard, Menlo Park, CA. 1975. (Available from *Creative Computing* Library. See ad.)

This is an interesting and timely book. At all levels it represents a melding of diverse, even polarized elements. At once it is a potpourri of games, an activity dating back to the roots of man, and a catalog of creative ways to use man's most modern and sophisticated tool, the computer. The contents range from old number games like NIM, which originated in ancient China, to STTR1, a simulation adapted from a modern science fiction television series. The computer version of NIM (and many of the other games) is merely an adaptation of an age-old diversion; these games may be enjoyed equally well without the use of a computer. Games such as STTR1, on the other hand, would never have been possible without the speed and computational abilities of modern computers. (In fact, STTR1 not only requires a computer to be played; it includes a computer as an integral part of the game.)

The publishing of this book represents a combined effort. The computer games and simulations were contributed by a variety of people, from mathematicians to businessmen, from students to professors, and from computer phreaks to homemakers. Indeed, the book was made possible through the joint cooperation of the People's Computer Company, a store-front, non-profit, educational group that publishes a funky newspaper, and Hewlett-Packard Corporation, a more traditional company that manufactures time-sharing computers, among other things. This, in itself, is significant.

In short, *What To Do After You Hit Return* is destined to become one of *those* books . . . It is conspicuous—one of those books that is too big to fit on the shelf, so you find it lying about on a table; it is eclectic—one of those new, soft-cover newsprint catalogs that is crammed to the margins with interesting tidbits and graphics; it is a curiosity—one of those books you feel compelled to pick up, just to see what is inside; and most important, it is an educational resource—one of those books that will help you find, obtain, or "get into" new materials for the enrichment of learning.

Bob Kahn
Lawrence Hall of Science
University of California, Berkeley

*****

Game Playing With Computers, by Donald D. Spencer, $12.95. Spartan Books, New York, 1968.

This unusual book, completely devoted to computerized game playing, introduces more than 70 games, puzzles and mathematical recreations that may be programmed for a digital computer.

Over 25 complete programs are presented, most of which have never before appeared in computer books. Each of the programs includes a description of the game, flowchart, a program written in FORTRAN, BASIC, a description of how the program works, and output produced by the program, In addition to an in-depth analysis of over twenty different types of magic squares and how they may be generated by a computer, the book features complete programs for prime numbers, Sieve of Eratosthenes, pick-a-number, blackjack, binary card games, the counterfeit coin game and 15 other puzzles.

To stimulate the interest of students and beginning programmers and to challenge the ingenuity of more experienced senior analysts, senior programmers and mathematicians, more than 50 games are presented for computer solution. These include such popular pastimes as Keno, Roulette, Go, Chess, Nim, Pantomino and Tic-Tac-Toe.

*****

Games, Tricks and Puzzles for a Hand Calculator, Wallace Dudd, 100 pp, $2.95. Dymax, Menlo Park, CA, 1974. (Available from *Creative Computing* Library. See ad.)

Any young person who has access to an electronic calculator of any description will find something in this book to challenge and amuse him or her. A child just learning the elements of arithmetic will find games and tricks at the appropriate level. At the same time, there is at least one game that is sufficiently difficult to analyze so that a trained adult mind will have difficulty in finding a winning strategy. (The reviewer admits to not having solved it. Our solution appears in a separate article in this magazine.)

There is a rapidly growing literature on the hand calculator, but I find this book outstanding because of its attractive presentation and wealth of ideas. For example, there is a good chapter on isolating malfunctions, which might have been subtitled "How to Become the Sherlock Holmes of the Calculator Age." All in all, well written and a lot of fun.

L.D. Yarbrough
Lexington, Mass.

*****

Games Calculators Play. Thomas J. Seymour, 32 pp, $2.00. Seymour Publications (P.O. Box 1788, Rockford, IL 61110), 1975.

I don't know who first turned his calculator around with 0.7734 on it and noticed it now said "HELLO", but immediately a new pastime was born: calculator word games. Unfortunately the vocabulary is limited to I,E,H,S, L,B,O,G, and D depending a bit upon the calculator and your imagination.

This little book is a collection of 57 simple but clever calculator problems similar to the following: "If a man invests $211,843 in the stock market (ENTER 211843) and his stock prices fall 26 percent (ENTER x 26), what does he have?" (Answer: BIG LOSS) Also included is an appendix of some 250 words possible to form on a calculator. Lots of 5379919.

David H. Ahl
Morristown, NJ

*****

Mathematical Carnival, Martin Gardner, $8.95. Alfred Knopf, New York, 1975.

So everyone has a pocket calculator. You casually remark at a party that you don't need one of these toys because you can multiply two nine-digit numbers in your head. Ha, ha. Someone challenges you. "Get pencil and paper" you say, "and I'll give you the one nine-digit number and you give me the other." You write 142,857,143, your friend writes, say 123,456,789. Without hesitation you take the paper and write 17,636,684,160,493,827. He takes it off to a corner and after 5 or 10 minutes is back looking at you with more awe than before. Incidentally, most small pocket calculators can't handle this problem—too many digits.

Gardner explains this trick along with scores of others such as naming the day of the week for any specified date. He also describes the work of Dr. Fliess, a Berlin physician who was talking about biorhythms (male 23 days, female 28 days, he believed) way back in the 1800's. Included too are a variety of games including the particularly intriguing "Sprouts."

This book has something for everyone, beginner and enthusiast alike. You'll enjoy it!

David H. Ahl
Morristown, NJ

*****

The Computerized Society. James Martin and Adrian R. D. Norman, 560 pp, $12.50. Prentice Hall, New York, 1970.

The Computerized Society is a lengthy book of 29 chapters, divided into three sections which discuss what have become the traditional topics of computer and society. Attention is given to the rapid growth of computing in the last 30 years, present application areas, the social and philosophical consequences of the new technology, and the controls which may be necessary to insure the proper use of computers.

The book ranges from well-documented chapters on the growth and uses of computing, to the authors' rather nebulous attempts at formulating policies which would lead to a Utopia with the new technology. Too much attention is given to categorizing mankind and then assigning jobs based on the often questionable results of IQ tests.

One problem with books of this type is that they become dated almost before they get into print. For example, the quote opening the chapter on Law Enforcement: "When little old ladies have to wear tennis shoes so they can outleg the criminals on the city streets, there's something wrong with the way we're doing things" is by Spiro T. Agnew. Certainly a better choice could have been made.

The book does give a thorough treatment of all the topics important to computers and society, and, in general, is useful background material for the professional in computing or the layman concerned about the implications of the so-called "thinking machines". Unfortunately, given the book's shortcomings, it cannot be recommended for general reading.

David L. Feinstein
University of Wisconsin
River Falls, Wisc. 54022

*****

Simulation Games in Learning, Boocock, S.S. and Schild, E.O. (Eds.). Sage Publications, Beverly Hills, CA. 1968.

In 1968, when *Simulation Games in Learning* was published, simulation games—the combination of the ancient technique of gaming with the relatively recent technique of simulation—were viewed as an educational innovation. When published, the purpose of the book was to present a progress report on this new technology, i.e., present a "valid picture" of what is known today about simulation games in learning and of current work in the area. The volume actually grew out of two issues of the *American Behavioral Scientist* edited by James S. Coleman.

In 1975, the book has very little new information to offer the educator interested in instructional computing. Most of the research findings have been reported or summarized in more recent publications. (See, for example, *Simulation and Gaming in Social Science* by Inbar and Stoll, 1972, or *Simulation Games for the Classroom* by Mark Heyman, 1975). In addition, only one of the seventeen articles deals with computer-based simulation games. In that article, Richard Wing describes a small-scale, controlled experiment designed to examine the applicability and learning effectiveness of two simulation games—the Sumerian Game and the Sierra Leone Game—with sixth grade students. The results of the experiment led the author to conclude that ". . . computer-based games can be used in practice even with sixth graders; they do teach as well as conventional classroom methods; and they seem considerably more effective than conventional methods, when the time investment of the student is taken into consideration."

Individuals that have used or run some version of the Sumerian Game (SUMER) may find this single article interesting, since it gives some information on how the simulation game is intended to be used.

Dan Klassen
Lauderdale, Minn.

*****

Math, Writing & Games in the Open Classroom by Herbert R. Kohl, 252 pp. $2.45. Vintage Books by Random House, New York ,1974.

This book is two small, loosely related works bound in a single jacket. The first part "Approaches to Writing" is partly a rehash of topics in the author's earlier book *36 Children,* and is based upon his experiences teaching writing and self discovery to children in a Harlem school, as well as conducting seminars for elementary school teachers.

Part II, "Games and Math", deals with simple, inexpensive games for young children which were created by both adults and children and which help make learning about science, geography, music and mathematics more fun. Simple versions and modifications of games such as chess, checkers, nim, Go and Wari are illustrated and discussed. Since the book appears to have been written for the elementary school teacher who has not yet discovered educational games, most of the games are rather tame stuff for the game buff, especially the computer game enthusiast. However, there is something here for the expert, because Mr. Kohl categorizes and discusses games according to *themes, playing boards, pieces, decision devices, goals* and (most important) *how children learn to play games.* He also illustrates the educational value of games by showing how skills and attitudes can be developed through game playing, how games can be made a part of all areas of the curriculum, and what sources teachers can draw upon for ideas and resources for games. The distinctive approach to educational games in this book is that children should be encouraged to create their own games, to modify the rules of standard games to suit their own purposes, and to play games because they want to, not because it is required by the teacher.

The author discusses the similarities between the way theories are constructed in natural and social sciences and the nature of creating, exploring and modifying games. When creating games, children construct and explore theories, perform experiments, make appropriate modifications, and examine the consistency and applications of their theories. Games are shown to be good ways for children to learn how to work together and to make decisions.

While most readers may find a few new ideas for educational games in this book, the author's major purpose seems to be that of changing the reader's view of education and educational games and to present some of his own opinions about teaching and learning. This book is nontechnical and informal, and the general reader may find Mr. Kohl's approach to games and other teaching strategies in the "open classroom" interesting and informative. However, there is no mention of computer-based games.

Frederick H. Bell
University of Pittsburgh

*****

Getting the Most out of Your Electronic Calculator, William L. Hunter. $7.95 Hardbound, $4.95 Paperback. TAB Books, Blueridge Summit, PA, April 1974.

So Uncle Fred has given you a handy-dandy pocket calculator for Christmas—or perhaps you are contemplating getting one for yourself or a friend. What now? Is it really worth the money, and just what is it good for, except as this season's "in" toy?

This book is an attempt to answer some questions about what the little boxes are for and how to make some intelligent use of them. In so doing, it points up the amazing variety of hand calculators on the market and their differences and similarities.

The strongest points about the book are, first, it is about the only such book now on the market, and second, it has some interesting chapters on applications, especially on Income Tax preparation. Of particular interest are photos and manufacturer's blurbs on 27 different models. However, this portion of the book is already obsolete because of the dramatic drop in prices and the introduction of new models in the year or more since the book was compiled.

My evaluation of the book is largely negative. It is obsolete, as was mentioned above; although it is intended "to serve as a modern course in general and commercial mathematics", it does not seem well suited for its avowed purpose (it is filled with examples, but there are no exercises by which a student could measure his skills); it is far better designed as a reference than as a text book.

Finally, the examples are worked out in terms of a non-existent "typical" machine so that each algorithm must be translated into the framework of the user's particular capabilities. On the other hand, if you are buffaloed by your own calculator and feel you are not getting your money's worth, you ought to check this book out. It may be just what you need to get you started using it effectively.

L.D. Yarbrough
Lexington, Mass.

*****

Problems for Computer Solution, Fred Gruenberger and George Jaffray, 401 pp, $7.25. John Wiley & Sons, New York. 1965. (Available from *Creative Computing* Library. See ad.)

A superb selection of 92 problem situations from business and science as well as mathematics. Each problem is well described and specific exercises are identified.

Numerous settings are presented on prime number applications, decimal representations & probability related problems using random numbers. Many business examples are described including compound interest, dividends, sorting & statistical procedures (queueing, curve fitting, quality control).

Although finished programs are not presented, the problems and suggestions are clear and thorough.

Hank Kepner
Milwaukee, WI

*****

Primer in Computer Utilization. Richard F. Curtis and Maynard L. Erickson, 233 pp., $6.95. General Learning Press, Morristown, NJ, 1974.

Fortran IV is introduced by the authors with a non-mathematical flair. An attempt is made to develop in the reader a basic level of programming skill by stressing the logic of programming, giving experience in writing simple programs, and mastering each level of learning before progressing on to the next level.

The reader is led through numerous problem solutions aided by diagrams, sample program printouts, and graphic illustrations.

A social science, psychology, pre-medical, or other non-mathematics student may find this primer a refreshing first exposure to the realm of computing at the undergraduate level.

-Gary D. Schafer
Lauderdale, Minn. 55113

*****

Computer Algorithms and Flowcharting. Gerald A. Silver and Joan B. Silver, 170 pp., $5.95. Gregg/McGraw-Hill, New York.

The authors lead the reader through the steps necessary to logically analyze a problem and formulate it in a fashion which is readily computer-programmable.

The topics covered can be loosely divided into four units. The first is devoted to problem analysis and algorithm development. The second unit involves flowchart preparation, including what is sometimes called macro and micro flowcharts, and a very detailed presentation of specialized flowchart symbols. Programming techniques such as conditional and unconditional branches, loops, counters, and arrays are treated in depth in a third unit. A final unit on applied programming logic involves presentation of real-life business situations and their solutions using the techniques previously discussed.

Each section in the book contains a set of exercises. The early exercises emphasize the language of computer programming. Later exercises reinforce key terms and provide practice in developing computer algorithms for solutions of problems. The text contains an abundance of clearly presented flowcharts that clarify the programming techniques and solutions to problems. Two nice points about the book are that no previous programming skill is needed, and that the book can be used with a variety of languages such as Fortran, Cobol, and Basic.

This well-written book would appeal to the business department rather than the mathematics department in a school. I would recommend this book for use in a data processing course with access to a computer, or as a reference book to be used for its description of flowcharting and problem analysis in the business field.

Bruce W. DeYoung
Oakland, N.J.

*****

Mathematics, A Human Endeavor. Harold R. Jacobs. 529 pp. $8.50. W. H. Freeman and Co., 660 Market Street, San Francisco, CA. 94104.

Mathematics in the Modern World. Readings from the Scientific American with introductions by Morris Kline. 409 pp. $7.00. W. H. Freeman and Co.

Although one doesn't have to know much mathematics to deal with computers, it doesn't hurt to know some. Here are two very attractive books (W. H. Freeman certainly knows how to put out a good looking book) that introduce various mathematical subjects to readers or students who may have missed them (or slept through them) on the way through school.

Jacobs subtitles his book "A textbook for those who think they don't like the subject" and although I happen to like the subject, I think it is a good book for those who don't. But it's good for people who like math too. His chapter titles suggest what his book covers: 1. The mathematical way of thinking, 2. Number sequences, 3. Functions and their graphs, 4. Large numbers and logarithms, 5. Regular Polygons, 6. Mathematical curves, 7. Some methods of counting, 8. The mathematics of chance, 9. An introduction to statistics, 10. Some topics in topology. Jacobs style and approach are charming. He keeps asking students and readers to think instead of just read. His book is full of good writing (His introduction to logarithms is a gem.) It is full of cartoons (Peanuts and others), advertisements, photographs and good diagrams. There is very little here that bears directly on computers but lots of things that people who deal with computers could use. This book is suitable not only as a high school or college text but also to read just for the fun of it. Highly recommended.

"Mathematics in the Modern World" is a book of readings from the Scientific American. Here are articles by Halmos, Kline, Davis, Euler, Kac, Quine, Dirac, Einsein, Ulam, Kemeny and other biggies. Among the articles directly relevant to the computer field are articles on "Mathematical Machines" (Davis), "Computers" (Ulam), "Computer Logic and Memory" (Evans), "The Uses of Computers in Science" (Oettinger), "Systems Analysis and Programming" (Strachey), "Cybernetics" (Wiener), "Man Viewed as a Machine" (Kemeny) and others.

Readers familiar with the Scientific American will know what to expect. Here are well edited and well illustrated articles that no normal person can quite use to learn about a field but which will get somebody started and will be appreciated by experts for saying a lot in a little space, and saying it both well and correctly.

The reader who plans to buy this book should note that there are two other books of readings ("Information" and "Computers and Computation: Readings from Scientific American") that also collect articles from the Scientific American about computers whose articles tend to overlap the ones in this book. Also Freeman sells reprints of articles from the Scientific American individually. Still this collection is very attractive and reasonably priced.

Peter Kugel
Boston, Mass.

CREATIVE COMPUTING Reviews

SPACE:1999

By Ruth Glick
Columbia, Maryland

As a substitute for *Star Trek* reruns, this season's non-network science fiction series called *Space 1999* just doesn't make the grade

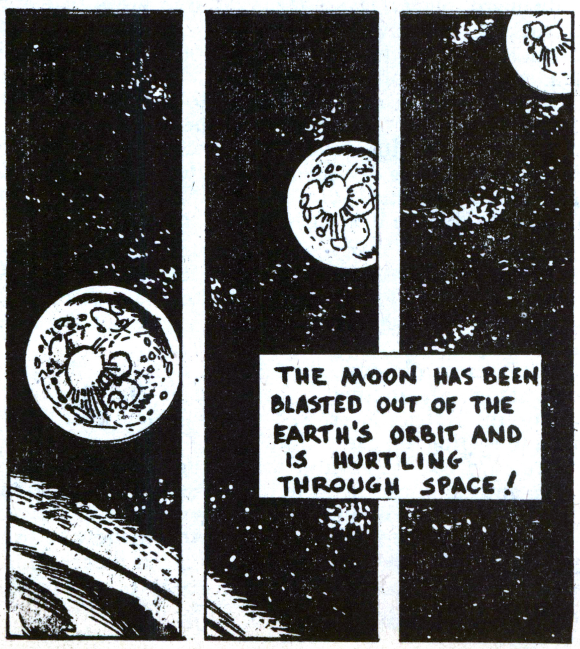
How come? Hasn't it got spectacular special effects like the moon blasting out of orbit, plenty of action, and Moon Base Alpha uniforms designed by Rudi Gernreich?

True. But these are only superficialities—the external trappings of T.V. science fiction. Unfortunately, *Space 1999* also has infantile plots, boring characters, ridiculous motivation, poor story construction, gadgets for their own sake and a whole wax museum full of unexplained and unbelievable monsters.

Comparing the new show directly to a successful series like *Star Trek* is a good way to illustrate the problems. Let's look at a memorable *Star Trek* monster—the horta—for example. It wasn't just a mindless beast menacing a group of miners. During the show, its intelligence was established. And later it was given believable motivation—that of a mother defending her young.

On the other hand, *Space 1999's* second episode treated the viewer to an old-fashioned horror show featuring an oversized hydra that sucked in human victims and spit out their charred remains. But no attempt was made to explain what the monster was, where it came from or why. Similar objections can be raised to the other monsters *Space 1999* parades on the T.V. screen almost every week. Each is simply a *deus ex machina* brought in to foster a particular story line.

Lively characterization is another missing element in *Space 1999*. One of *Star Trek's* biggest appeals was the crew of the star ship Enterprise. Spock, Captain Kirk, Dr. McCoy, Uhura and the rest still have a passionate following today because of their well-defined, complicated personalities.


THE MOON HAS BEEN BLASTED OUT OF THE EARTH'S ORBIT AND IS HURTLING THROUGH SPACE!

The staff of Moon Base Alpha also comes in an international assortment of colors. But there's no one the viewer really wants to root for. The female lead, played by Barbara Bain, is colorless and wooden. And the other cardboard characters, with the exception of the base's Commander Koenig, are like interchangeable pawns moved about by the plot.

Koenig himself is another problem. Unlike *Star Trek's* Captain Kirk, who was allowed some human fallibility, he's never wrong. And his dependable infallibility makes it possible for the viewer to figure out very quickly where the plot of any given *Space 1999* episode is leading.

Of course, being able to unravel the plot quickly is a definite handicap, since the stories on *Space 1999* usually drag—with long slow scenes that barely advance the action. Take the space expedition at the end of which that over-sized hydra was lurking. For 10 minutes there was no action. To make the sequence even slower, it was told as a flashback, unaccountably narrated by Barbara Bain, who wasn't even present during the expedition.

Repetition of plot devices is another deficiency. One week Alpha is being menaced by a blinking blue light. The next week it's flashing a green light. And, in every episode so far the script dredges up the danger of radiation leakage.

It's possible to point out a lot more problems—especially with the show's hardware. First, there are those ridiculous little gadgets that the base personnel use to open doors. You'd think that man would have invented something better than a glorified key by the year 1999. Imagine if one dropped off someone's belt and he were trapped in a room for days before being missed.

Or what about the use of computers in *Space 1999?* Maybe they don't have the technology to match *Star Trek's* "library computer." But they should be able to do better than a machine that prints out its hard copy on what looks like adding machine tape.

And then there are the hand weapons—which are almost identical to present-day staple guns—except that they shoot energy rays instead of staples.

It's obvious that far less thought went into designing believable equipment for Moon Base Alpha than for the star ship Enterprise.

However, *Space 1999* does have it all over *Star Trek* in the special effects department. Their moon landings, space-suited lunar surface sequences, and atomic explosions are unbelievably good for a T.V. production.

But they don't have the scientific accuracy to match the effects. Isaac Asimov took about 1500 words pointing out these problems in a *New York Times* review of the show more than a month ago. His biggest quarrels were with the number of planets Moon Base Alpha has encountered in such a short time and with the show's inaccurate references to a "dark side of the moon."

With a highly successful model like *Star Trek* to crib from, the producers of *Space 1999* should have come up with some satisfying science fiction drama. Too bad that what they have is a bunch of spectacular special effects in search of a decent story line and a few interesting characters.

CREATIVE COMPUTING
Feature Review

34 Books on BASIC

Stephen Barrat Gray
Gray Engineering Consultants
260 Noroton Ave.
Darien, Conn. 06820

Installment Number 5

This group review of 34 books on BASIC started in *Creative Computing,* Vol. 1, No. 3. Five to seven reviews appear in each issue. More next issue.

For a future group review of books on applications of BASIC, I would appreciate information concerning such publications. This would include not only books such as Peckham's *Computers, BASIC and Physics,* but also applications books not oriented toward any particular language, but which could be used with BASIC, such as Gruenberger & Gaffrey's *Problems for Computer Solution.*

Also appreciated is information about books on BASIC in languages other than English.

22. *Fundamental Programming Concepts*, by Jonathan L. Gross and Walter S. Brainerd. Pub. Apr. 1, 1972, by Harper & Rowe, New York, N. Y., 304 pages, 6 x 9¼, $9.95 (hardcover).

Some excellent portions, but too many difficult areas unexplained, far too much extraneous or overly complex material. Rating: for classroom use, B+; for the solitary reader, C–

On the dust jacket, some of the letters of the title are filled in, generating a subtitle: Fun Programming. This book could be quite a lot of fun, in the hands of a good teacher who could explain the hard parts, and cut through the material that doesn't belong. But in the hands of the solitary reader, this text provides more exasperation than fun.

The preface says "This book is the appropriate length for a concentrated semester course and could also be used as a self-teaching guide for a motivated person who has a time-sharing service available." Highly motivated, that is, and with an IQ above 150.

There are ten chapters: Elementary BASIC; Advanced BASIC; Information Processing; Character Strings; Mathematical Methods (maxima and minima, approximating roots, area, graph plotting, matrix operations); Simulation; Two Puzzles (algebraic, colored cubes); Artificial Intelligence; Machine Language, Assemblers, and Compilers; Programming Languages.

The beginning is good, a five-line program that adds two constants, using three LET statements, on page 2. The next page has a five-liner that finds the product of three constants, using READ, DATA and LET.

Page 6 begins a section on using a terminal, with 11½ pages on remote access, and on writing and running programs. The writing is simple and straightforward, with a rare sense of humor: "The computer does not have extrasensory perception, so it cannot detect errors of meaning, like adding the wrong number or naming the wrong variable." Or later, "On some systems the more euphemistic word UNSAVE is used instead of KILL."

But by page 20 the flaws begin to show. There are not enough examples; none at all in some cases, as in the paragraphs on relation symbols. Flowcharts are introduced early, but there are only 13 in the whole book.

The first major flaw is on page 25. Line 130 in the PRIME program is not explained; it is not readily apparent why I is compared with N-1 rather than just N.

If this were the only unexplained program line, then little harm would have been done. But there are many such puzzlers that only a professional programmer could easily understand, and which leave the beginner confused and exasperated. On page 73, line 340 of the BOOK program reads PRINT B;"   "; M;D without an explanation that quotes around blank spaces are one way of formatting output. On page 58, BKWRDS prints 20 numbers both forwards and backwards; there is no explanation of a key line, 190 FOR I = 1 to 10, although there are 20 items in the list, not 10. This is the introduction to subscripted variables, which need not be so complex, with 23 lines, and with one line that raises more questions than the program answers. Lines 200 to 220 are also not explained, other than in the REM line, EXCHANGE ELEMENTS. In general, there is not enough explaining of programs, usually only a general description, no explanation of the lines that are different from what has been previously encountered, except in a few very rare instances.

There is a group of problems after most of the sections in each chapter, although without answers. Some of these problems are for high-IQ types only, such as the ninth one of page 45, which asks the reader to "write a program that allows a person to play the game of matchsticks against the computer."

In many places, the authors get into areas that are off the track. In the example given for GOSUB and RETURN, the book goes off into an explanation of a return-address stack. Although some readers may want to know what's inside the machine, it is extraneous to the subject at hand, as is the section on "subdivisions of the scratch area."

By page 54, the reader has become conscious of a writing style that is dry and often stilted: "It is emphasized that the computer creates the data stack during compilation, before it begins running the program."

There are too many tricks that may only confuse the uninitiated, such as using the line READ X,Y,X,Y,X in explaining RESTORE. This is fine for the high-IQ readers curious about the byways of BASIC, but not the average reader.

A 37-line program that computes the charges for long-distance telephone calls is used to demonstrate double-subscripted variables. Not too difficult, but why use such a long program every time?

By page 69, the authors are through describing BASIC, and go into applications; the problems get much harder in this part of the book. Problem 2 on page 76 asks the reader to use data such as 10010, which must somehow be broken down into the individual binary digits, but there is nothing up to this point (or anywhere else in the text) to indicate how to do this.

Many other things are left for the reader to figure out. The only explanation of how bubble-sorting works is one brief sentence and one REM line.

The authors are stingy with examples. Built-in functions, such as ABS and SGN, are introduced without any examples of why, how and when to use them. There is only one example of a table. There are no examples of PRINT with and without comma and semicolon.

Most of the application programs are much too long and complex to be useful to the beginner who has learned what BASIC he knows from this book. There is little point in using BOOKS, a long library-book-availability program that requires such a long explanation, and which does little if anything to teach BASIC. There are seven pages on cryptography, which must be a favorite subject of one or both authors; this is usually and end-of-book subject, but here it is used as an application of strings, for other than which it is rather trivial. Page 173 begins four pages of explanation for an archeology program. Only one line is explained in a 99-line program on baseball simulation.

The most curious program is in Appendix C, which contains "a complete description and a listing of the BRAIN simulator," a "mythical computer" introduced to help in "developing a little programming skill in machine language." This is quite interesting, but certainly does not belong in a beginning text. It requires 24 pages in the main text, and all ten pages of Appendix C, including a three-page flowchart and a four-page listing of the 285-line program.

The title of the book should be more on the order of "Beginning BASIC with Applications." However, this is a good book for advanced work, or for browsing in the applications portion.

There are some unique portions. These are the only authors to distinguish between the use of parentheses and brackets in BASIC, noting that "arithmetic expressions use parentheses to indicate the order of evaluation. Function arguments are enclosed in parentheses. Subscripts and dimensions are enclosed in brackets. However, many versions of BASIC allow the programmer to use parentheses and brackets interchangeably."

The authors are the only ones to say, "Usually, the main concern with a BASIC program is getting it to run successfully, not necessarily most efficiently. Therefore, a higher premium is placed on clarity than upon brevity." They are alone in showing how to generate random numbers "if there is no random-number generator in the language or if the one available is not satisfactory," and in giving "suggested reading" throughout the text, after many of the groups of exercises. They are among the very few to get into the economics of programming: "The cost of computing is going down and the cost of programming is going up, which means that there are fewer and fewer occasions which justify the use of assembly language programming." They are among the very few who discuss the infinite loop. There are five fine pages on debugging, going into hand simulation, tracing with PRINT statements, and sample runs.

The last problem in the book is a short one, "Write a program that will produce rock and roll music." How can a beginner do this after reading only one page on "a program that composes music"? This is very reminiscent of the last section in Kemeny & Kurtz (2), "Harmony in Music," which has over three pages on a more difficult area of music, with a long program and also a long and careful explanation of it. The last problem in the Kemeny & Kurtz book is much more reasonable: "Try the program MUSIC on several melodies and suggest improvements in the program based on your experience."

Chapter 10, on Programming Languages, has a unique and interesting feature: an 8-line BASIC program is presented, followed by discussions of its equivalent in FORTRAN, ALGOL, and PL/1. That's fine, but why follow it with eight pages on SNOBOL?

In trying to be all things to all readers, the authors end in exasperating most of them.

* * *

23. *Programming Time-Shared Computers in BASIC*, by Eugene H. Barnett. Pub. Apr. 14, 1972, by Wiley-Interscience div. of John Wiley & Sons, New York, N. Y., 366 pages, 6 x 9, $12.00 (hardcover).

Packs much information into a well-designed book, with many unique features. Rating: A

One of the best organized of these books, this is meant to be used (according to the dust jacket) in high schools, colleges, in-house courses, and for self-study. It should do admirably for any of these.

A short program is presented on page 4, after a discussion and a flowchart.

The subsequent presentation of programs proceeds slowly and surely, and although the program on page 40 is 61 lines long, it is a very simple program that is easily understood. This payroll program, which includes employees Mike and Deborah Barnett (nepotism or child labor?), is modified and enlarged upon up to halfway through the book, by which time it has taken on a very new look.

There are eight chapters: Introduction; Elementary BASIC Programming; Programming Hints; Matrices, Input, Random Numbers, and Other Good Features; Advanced Examples; Advanced BASIC Features; Data Files: Input/Output; Computer Features. The book ends with solutions to selected exercises.

There are several sets of exercises in each chapter, presented right after each major statement or group of statements. These exercises are given in a nice variety: at the end of the section on GO TO, for instance, there are exercises on interest, test-score averages, center of gravity, vectors, calculating area under a curve, and standard deviation, among others, and this by only page 60. Some of the exercises are quite far-out, such as the one that asks the reader to "computer the (x,y,z) position of a satellite," given half a dozen parameters.

By page 73, after IF-THEN, the author figures it's time for the reader to get a better idea of how problems are solved with the computer, so there are 18 pages of "more sample problems," with programs and full discussions on subjects such as loan interest, largest number, depreciation, moving average, linear regression. quadratic equations, and binomial probability.

Over 21 pages are devoted to A Bag of Tricks, which are "little gimmicks" of the type the reader is urged to "consciously develop." The author discusses first the accumulator, which he believes to be "the most important single trick in computer programming," then goes into counters, interactive techniques, recognizing the end of data, step functions, and several others. One of the best is that of using employee numbers as line numbers, and then letting the computer sort them into numerical order.

This is the only book other than Sass (21) to use MAT arithmetic for business applications, such as accumulating total sales for each retailer, salaries, production and distribution methods, product test data, etc.

Random numbers is an area to which many authors devote a minimum of space; Barnett has eight very nice pages.

By page 231 the author decides it's time for advanced examples, and presents 24 pages of problems, programs and discussion, covering, among others, moving averages, sorting, survey analysis, simultaneous equations, Monte Carlo simulation of a business decision, random walk, and even simultaneous differential equations, a nice batch of sophisticated applications.

The last chapter contains one of the few sections on editing in these books, and the only one to discuss time-sharing contracts.

A unique feature is a preview, right up front on page xiii, telling the reader what he can look forward to, with a list of Program Subjects, "The following major subjects are illustrated by various examples and exercises throughout this text." There are 54 subjects listed, under the headings of business and financial, sociology, engineering, mathematics and operations research, and miscellaneous. A very nice pre-sell to show the reader what big, exciting things are ahead of him.

There are very few things to say against this book, and all are relatively trivial. A few statements are somewhat confusing, such as "An important aspect of the READ statement is that the computer has the ability to never read the same DATA entry twice (except under special conditions . . . .)." True, but the wording is nevertheless confusing. On the same page, ". . . the READ statement can read as many variables simultaneously as desired." Simultaneously?

Barnett is the most realistic of all these authors. In a forward to the first set of "more sample problems," he writes: "It is the student's immediate task to: simply understand how each program does, in fact, implement each problem; begin to understand some of the trickery, chicanery, and subterfuge used to accomplish real-life tasks . . . ."

* * *

24. *Introducing BASIC,* by Theodore R. Blakeslee, II. Pub. Aug. 1972 by Educomp Corp., 298 Park Road, West Hartford, Conn. 06119, 162 pages, 8½ x 11, $4.95 (paperback).

Although there are many good parts, the coverage is uneven and the writing often stilted. Rating: C

Although this book is not published by a book company, it was, according to the preface, "written as a general primer for BASIC, and may be used easily with any computer system supporting the BASIC language, [although] the book was designed around the Edusystem line of computers marketed by Educomp." At the time this book was written, both Educomp and Digital Equipment Corp. were using "Edusystem" for their computer systems designed for schools. Then DEC copyrighted the name EduSystem, so Educomp now uses EDUCOMPUTER.

There are eleven chapters: Introduction, Fundamentals, Communicating (INPUT, PRINT, TAB), Assignment Statements, Control Statements, Arrays, Functions, Subroutines, Teletype Graphics, Errors, and "Pity the Poor User." There are exercises at the ends of chapters 3 through 10, without answers.

The preface notes that "Emphasis is placed on two important aspects, the random number generator and terminal graphics." Actually, the emphasis is on Teletype graphics (a 20-page chapter), not on the random-number generator (three pages in the chapter on functions).

The cost of the book has been kept down by typing the text, leaving the right margin unjustified, and doing all the artwork freehand (lines around boxes, exponential arrows, etc.). So the cost has been kept below $5.00, but the appearance suffers. This is the only book without consecutive page numbering; each chapter starts with page number one, such as 7-1.

This book is not very well written; it tries to be conversational, but still has too much of the pedagogue, so the beginner won't find the going very easy in a number of places. On the second page: "The computer language called BASIC is, itself, a computer program. It is a very complex program written in a different kind of computer language ("machine" language), one that requires great detail when writing a program." This on page 1-2 of a book that is designed to "let you obtain a working knowledge" of BASIC "with a minimum of teacher assistance."

Page 1-3 notes that every statement starts with a line number, but gives no examples. Nor are there examples of commands, which "use no line numbers." When writing about string constants, this is the only author who refers to elsewhere in the book for examples. There just aren't enough examples, and in some places, none at all.

Many sentences leave unanswered questions and therefore tend to be cryptic, such as "You should also note that the characters "12345" in a string (remember the quotation marks) are not the same as the numeric constant 12345 (twelve thousand three hundred forty-five)." And on page 7-5, "In addition, much of this mathematics also gives the constant e raised to a power as the result of manipulations (mystical, magical, mathematical manipulations)."

That alliterative phrase is a part of a curious and often awkward writing style that includes, "A discussion of plotting (nothing undercover) is postponed until Chapter 9." In writing about quotation marks in a PRINT statement, "No data is processed, just character-by-character regurgitation." A confession: "The author has used both kinds of LET statements and prefers the implied LET but must confess to a highly developed sense of laziness." Pedantic: "The LET statement resembles the algebraic equation but its operation is one of evaluation and assignment." Confusing: "Try using just a leading comma and no space" (without explaining what a leading comma is). Cute (heading for a printout): "RANDOM NUMBERS (SPECIAL SALE)." Or, on page 11-3, ". . . for use by people not at all familiar with the mysterious art of programming."

Some of the explanations are not too clear: "The comma will control the position of the type element in 14-position line segments." The author is the only one to use technical terms such as *embed* and *hierarchial.*

This is an uneven book, too generous in some places (nine pages on precedence), much too skimpy in others (only one flowchart in the book, and that a "verbal" one).

Program 5.06 as shown will not produce the given output, because the STEP of 1/2 is missing from the FOR statement. And if this is supposed to be an enlargement of the previous program, the step should be I/2, not 1/2. Had the author included the needed explanation, he would probably have realized that he left out the step.

There is no mention of the standard table dimension; page 6-3 notes that a program using the given list would have to include DIM A(6). Program 6.01 has a DIM A(9).

The explanation of program 6.02 says "The first 10 numbers in line 360 specify how many numbers are contained in each group." Not true; they are one less than the required value. Later the text notes that line 340 uses B(I)+1 and asks the reader why, but never tells him. This is an important but unexplained point.

The section on bubble-sorting, on the other hand, is long and very well explained.

The author often gets too far ahead of the reader, with complex programs to demonstrate something that could be done much better with a short program. The author then fails to answer all the questions the complex program brings up by virtue of its complexity.

The chapter on Teletype graphics has sections on "more on the TAB function," a simple graph, plotting with negative values, using Teletype graphics to solve an equation, and plotting multiple functions, this last with a program of 124 lines, five subroutines and 29 GOSUB statements to plot three functions.

Chapter 10, on program errors, covers debugging, with a seven-page example of debugging a program. The last chapter, "Pity the Poor User," covers the documenting of programs with flowcharts, generalization of programs, and output format. The opening states: "This chapter will be regarded by some as a sermon and an unnecessary sermon at that. However, having examined many student programs . . . ."

The writing is often on the Tom Swift level: "With fingers crossed and a feeling of dread we type RUN – AND THERE IT IS!!" Or this one, "A dummy variable, Z, is introduced, the changes entered, and one more hope rides high."

Appendix A is on Teletype terminals; Appendix B lists the BASIC statements, edit and control commands, and functions, and which of the five EDUCOMPUTERS uses each.

There is much emphasis on neat, simple and understandable program output.

* * *

25. *Computing With the BASIC Language,* by Fred Gruenberger. Pub. Aug. 1972 by Canfield Press, San Francisco, Calif., 140 pages, 7 x 10, $5.95 (paperback).

Good for reading after one or two others on BASIC, or after having some programming experience. Otherwise, too much extraneous (although interesting) material. Rating: as a first book; C–; as a second or third book, B

A disappointing book, when compared with others by Gruenberger. Although there are many interesting sidelights, they don't belong in a 140-page book "suitable for a one-semester introductory course in computing."

On the plus side, these sidelights are of most interest to someone reading this after having gone through one or more other books on BASIC, or having had some programming experience. The author is the only one to tell how to find out if the program is in an endless loop. He explains in more detail than most why the conversion from decimal to binary is not always exact. He gives the title and author of the first article on flowcharting. He is the only author to show what should *not* be done in flowcharting.

Gruenberger is the only one to show how to make a timing run. He alone asks the reader to run a program that will demonstrate the range of some of the built-in functions. He is alone in asking "What should we compute?" – and takes 3½ pages to answer. There is a fine section on "The Limitations of BASIC." He has one of the few write-ups on machine, assembly and high-level language. The first appendix gives three excellent pages on the internal workings of a BASIC time-sharing system.

The main text is set entirely in a typewriter font, with headings, programs and captions in five other kinds of type, resulting in an appearance that is less than handsome. There are exercises at the end of each chapter; the last exercise is number 310, but there are not 310 exercises; they are numbered 5, 10, 15, 20 . . . on the basis that this provides "for the insertion of new exercises"; there are actually 62 exercises.

The exercises are not set off from the text; on page 26, for example, the exercise runs into the text, and so this portion of the text, for one, could easily be skipped over if the reader is not careful. Actually, the reader may not always be able to tell what is text, or part of an exercise. If the latter, then the exercise contains important information that should be in the text, so as not to be missed by the hurried reader who skips the exercises.

There are eleven chapters: A First Look at BASIC; Data, Numbers and Variables; Flowcharting, More About Writing Programs; Looping; Functions; Subroutines; Problem Solving; Debugging, Testing and Good Computing; Extended BASIC Features, and A Final Word (The Elements of Computing, Your Future in Computing). There are six appendixes: About BASIC Itself, Things to Check For, Error Messages, Library Programs, Eight Advanced Problems (perpetual calendar, elapsed days, change maker, etc.), and a Commentary on Selected Exercises, with answers to only a couple of them.

There are only 97 pages of main text, so the information has to be poured on, with no space for investigating in detail, yet the author keeps straying from the subject at hand, going into areas that should be beyond the scope of such a text.

On a sink-or-swim basis, the author starts right off with an 11-line program on page 2, followed by a paragraph of explanation for each program line. This program would be difficult for many readers to understand.

Gruenberger is another author who emphasizes the nicety of not allowing a program to end in OUT OF DATA ON LINE XXX, saying this is "not the most graceful way to terminate a program," adding "but it works and is commonly used."

In several places he is cryptic: ". . . powers of ten and powers of two are incompatible," without further explanation. Some programs contain lines that are not explained at all, but whose importance is far from obvious.

The text begins to get a little murky on page 20, at which point the author gets into probability to illustrate flowcharting, with an example that is neither interesting nor simple.

As an example of the terseness of this text, the description of formatting with comma and semicolon takes only four sentences and two program lines.

The author writes as though he had first made an outline, and then filled it in as sparsely as possible. The style is rather dry, with only an odd item now and then to brighten up the desert.

This text seems to be another example of an expert writing for beginners without realizing that he's writing over their heads, that they may know absolutely nothing about programming.

Gruenberger doesn't seem comfortable in BASIC. For one thing, on page 36, he starts using the DO concept in describing a loop.

The chapter on Problem Solving with BASIC begins with bracketing, which is too much for many beginners, and is poorly explained, as are the three pages on Euclid's algorithm, which are very confusing; only a very clever reader could follow the text here.

* * *

26. *Business Programming With BASIC,* by George Diehr. Pub. Oct. 20, 1972, by Becker and Hayes, subsidiary of John Wiley & Sons, New York, N. Y., 344 pages, 8½ x 11, $8.95 (paperback).

A few good features, but mainly a jumbled, distracting mixture of solid masses of text and too many interrupting problems. Rating: C–

The basic idea of this text sounds good: it has a "semi-programmed" format, with questions throughout each chapter, usually related directly to the text. They are multiple-choice and fill-in questions, with the answers at the right side; the author suggests covering the answers with "an opaque card while you read the question."

But the questions come so thick and fast, interspersed in the text, that it's often difficult to figure out just which is which. The questions have a line before and after them, but there are often so many on a page that the separation isn't all that obvious.

Chapter 4, for example, is 23 pages long, followed by a five-page review quiz, and then four pages of quiz answers. Of the 23 pages, only 13 are text; the rest is taken up with 43 questions and answers. The entire book contains 330 questions in text, and 66 review-quiz questions. Some of the latter provide space for writing in the answer, so some pages have only a couple of lines on them.

Each chapter begins with a half-page outlining the chapter's contents. There are eight chapters: Computers and Terminal Systems; Introduction to BASIC; Conditional Transfers, PRINT Statement, Iteration; FOR-NEXT, Library Functions; String Manipulation, Input/Output Extensions; Subscripted Variables; Advanced Uses of Lists and Tables; Subroutines, Simulation. The three appendixes are on: extensions to BASIC, from file processing to PRINT USING; a summary of BASIC; and a table of system-dependent features and their parameters for nine time-sharing systems, from the B5500 to the XDS 5,7.

The first chapter is rather messy, jumbling together a too-long program, a description of terminal use, examples on two systems, error correction, and too many questions and answers, plus two pages on commands and editing features, for both CALL/360 and GE MARK II.

The writing is pedestrian, and sometimes confusing, such as "A character is one of the letters from our alphabet, a single digit, or one of several symbols such as * + = . . . Roughly speaking, when a number is stored in memory, two digits can be stored in the space required for a single character."

The text is typewritten, in a type so small that it isn't easy to read, especially since there are some very solid masses of type, seven inches wide and several inches deep. This is perhaps the worst feature of the book: the solid chunks of small type.

The author gets off to a bad start on the very first question, on page 3, asking who developed BASIC. The choices are: IBM, Kemeny and Kurtz, or "the creators of Screaming Yellow Zonkers." The rest of the 396 questions are all serious, but this first one sets an unfortunate tone.

There are 19 removable pages at the end of the book, to be taken out "so you can refer to one page while answering on another page." Since a duplicate of each of these is included in the relevant chapter, the redundancy seems unnecessary.

The first chapter presents READ and PRINT, but then, before making use of either, goes into a page and a half about a problem involving discount, interrupted by arithmetic questions intended to test the reader's knowledge of basic mathematics.

The first program is nine lines long, on discount off gross cost, with a fairly good explanation of the various steps involved, but rather long for a first program. On this same page the author goes right away into compilation, with a long paragraph on source programs, compilers, and object programs.

Four pages are taken up to show the discount program (now called a billing program) as run, corrected, saved and unsaved on *both* the IBM and GE systems, which seems wasteful.

There is some confusion as to terminology: on page 43, "After each line-number there is always an 'English' word. This word, called a 'Command,' describes the function or purpose of the statement." This author is the only one to use the terms "destructive read-in" and "non-destructive read-out" in describing the use of READ, PRINT and LET, and is the only one to refer to dummy data as a "trailer" value, to indicate end of DATA.

The text constantly refers to "figures" that explain various points such as constants, DATA, variables, etc. But these "figures" are just plain ordinary text with a line above and below, not drawings or charts at all, which is somewhat misleading. The style is rather dull, such as in this heading for the READ "figure": "Syntax and semantics of the READ statement."

The book's title is also misleading: the main text has very little to do with business programming. There are about two dozen complete programs in the book. Half a dozen are billing programs; the others are on assigning letter grades for a school, compound interest, optimal order quantity by search, finding values less than the mean, finding values less than the average, frequency tabulation, federal income tax, the "newsboy problem," etc. There is also a program that simulates a game of dice, and one that writes a song ("Old Macdonald's Farm") to illustrate the use of strings. Not much in the way of business programming in these programs, but there is in the two or three "programming problems" that accompany each chapter, which somehow seems a rather odd way of teaching business programming. As the preface puts it, "Each chapter concludes with a set of suggested programming assignments of varying difficulty. These assignments are designed to put the student's learning to the acid test. The programming assignments include data and correct answers for the problem." These programming problems are on computing unit price of supermarket items, monetary conversion, future values of investment, assigning letter grades, payroll program, depreciation, true annual interest, mean and standard deviation, rate of return on an investment, total receipts for a week, sales report, bubble sort, optimal replacement policy, maximum and minimum GPA, production costs, simulation program for inventory problem.

The sixth program is 44 lines long, but it's not too hard; it assigns letter grades depending on the student's numerical scores, and is mainly IF and DATA statements. The 17th program is quite complicated: 50 lines on an extended grading problem. Yet it does little other than determine the student's grade, and then list the various grades according to grade groupings.

This is one of the five BASIC books without an index, making it difficult or impossible to look up anything, except by spending much time searching the contents pages.

The jumble of text and questions is very hard to follow without getting lost or bored to death, or both. The book is needlessly confusing, and should have been heavily edited, set in book type, the padding removed, and the dull text improved. But would it be worth the trouble?

* * *

27. *Entering BASIC,* by John R. Sack and Judith L. Meadows. Pub. Dec. 1972 by Science Research Associates, Chicago, Ill., 133 pages, 7½ x 9¼, $4.95 (paperback).

One of the best, if the reader is very knowledgeable. But not enough detail for a beginner. Rating: for a programmer, A; for a beginner, C

A mixed bag, with some fine portions, and a preface that reads beautifully: ". . . BASIC is an ideal first step towards more complicated programming languages such as FORTRAN, COBOL, and ALGOL. Like the Roman god Janus who faces both ways, BASIC faces the needs of those just entering the world of automated data processing as well as those departing for its more stratified plateaus."

This book complements C. W. Gear's *Introduction to Computer Science* (also from SRA, 1973); "His book is frequently referred to for detailed discussions of specific points that go beyond the scope of this book. The two books are, however, mutually independent."

There are twelve chapters: Fundamentals of Computing, Interactive Processing, Introduction to BASIC, Assigning Values to Variables, Input/Output, Program Control, BASIC Functions, Subroutines, Array Handling, String Manipulation, Advanced Features, Debugging Aids. There are two appendixes: a Summary of BASIC Features, and Solutions to Selected Exercises. A 31-item glossary ends the book.

The book opens with a very readable style and a nice flow to the language. There is an excellent section, possibly the best, on Rules of Preference for arithmetic operators. The authors are among the few (9 of the 34 books) to introduce string constants and variables along with numeric constants and variables, rather than much later in the text, or not at all.

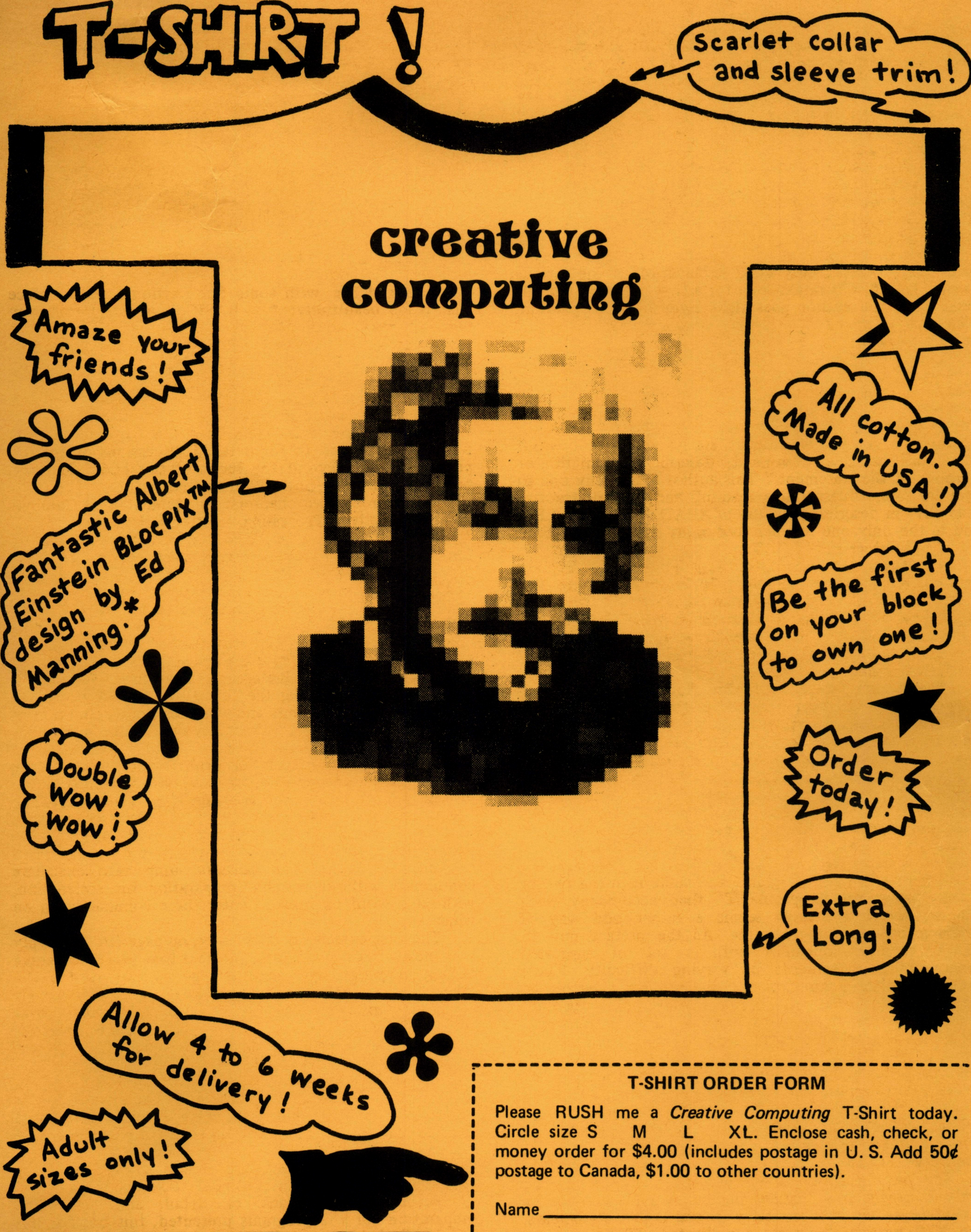
However, early in the book things begin to go a little too fast for comprehension, and without explaining terms. For example, "assembly language" is never explained adequately, in a section that would be difficult for most beginners, although not for a programmer. And yet the preface indicates that the book is for "the newcomer to computer science." The authors jump into executive commands without suitable preparation or explanation, without a word on how to enter these commands, or on what.

The first exercise is rather silly, on page 19: "To apply your newly learned wisdom, the next time you hear people talking computer jargon, casually drop the following: 'The other day I was communicating via teleprocessing lines with our time-sharing system in ————————. I logged on with the system executive, ————————, entered my program, and compiled it using the BASIC compiler . . . .'" Or are they pulling the reader's leg?

The first program, on page 21, is seven lines long, and prints the square roots of 1 through 5. The explanation of those seven lines is mixed in with a four-page general discussion of the elements of BASIC, and gets lost in the shuffle.

The program on page 39, on average rainfall, is too complex for many beginners to read; too much too soon.

Because of its lack of detail, and inadequate preparation for the programs presented, this book is not suitable for beginners, unless a teacher is available for filling in the gaps and answering questions. On the other hand, this is one of the best books for the knowledgeable reader, such as a programmer looking for a quick introduction to BASIC.


T-SHIRT !
Scarlet collar and sleeve trim!
creative computing
Amaze your friends!
Fantastic Albert Einstein BLOCPIX™ design by Ed Manning.*
All cotton. Made in USA!
Be the first on your block to own one!
Double Wow! Wow!
Order today!
Extra Long!
Allow 4 to 6 weeks for delivery!
Adult sizes only!

* BLOCPIX™ is a trademark of Watson-Manning, Inc., 972 E. Broadway, Stratford, CT 06497.

T-SHIRT ORDER FORM

Please RUSH me a *Creative Computing* T-Shirt today. Circle size S M L XL. Enclose cash, check, or money order for $4.00 (includes postage in U. S. Add 50¢ postage to Canada, $1.00 to other countries).

Name ____________________

Street ____________________

City ____________ State ____ Zip ______

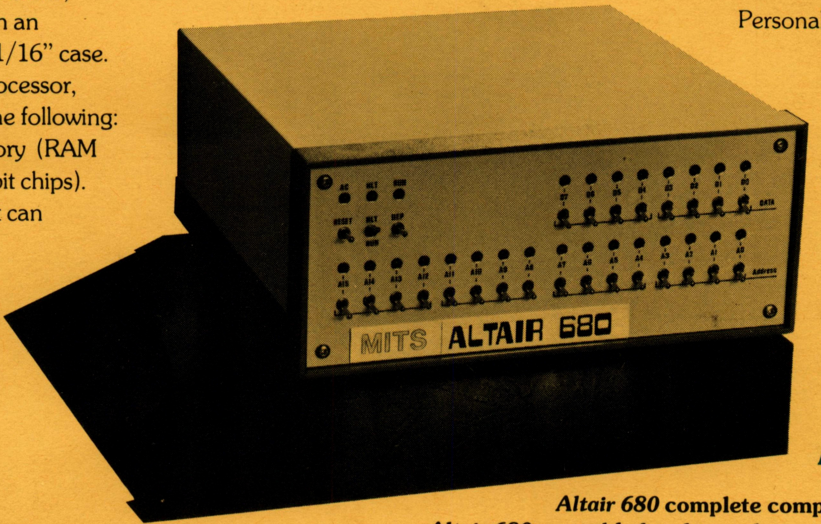
Return form to Creative Computing T-Shirt, P.O. Box 789-M, Morristown, NJ 07960 USA.

Lowest Price in the World!

In January of 1975, MITS stunned the computer world with the announcement of the *Altair 8800* Computer that sells for $439 in kit form.

Today MITS is announcing the Altair 680.

The Altair 680, built around the revolutionary new 6800 microprocessor chip, is the lowest priced complete computer on the market. **It is now being offered at the special, low price of $345!**



The Altair 680 comes with power supply, front panel control board, and CPU board inclosed in an 11" wide x 11" deep x 4 11/16" case. In addition to the 6800 processor, the CPU board contains the following:

1. 1024 words of memory (RAM 2102 type 1024 x 1-bit chips).
2. Built-in Interface that can be configured for RS232 or 20 mA Teletype loop or 60 mA Teletype.
3. Provisions for 1024 words of ROM or PROM.

The Altair 680 can be programmed from the front panel switches or it can be connected to a computer terminal (RS232) or a Teletype such as an ASR-33 or surplus five-level Baudott Teletype (under $100).

The Altair 680 can be utilized for many home, commercial or industrial applications or it can be used as a development system for *Altair 680* CPU boards. With a cycle time of 4 microseconds, 16-bit addressing, and the capability of directly addressing 65,000 words of memory and a virtually unlimited number of I/O devices, the *Altair 680* is a very versatile computer!

Altair 680 Software

Software for the *Altair 680* includes a monitor on PROM, assembler, debug, and editor. This software will be available to *Altair 680* owners at a nominal cost.

Future software development will be influenced by customer demand. MITS will sponsor lucrative software contests to encourage the rapid growth of the *Altair 680* software library. Programs in this library will be made available to all *Altair 680* owners at the cost of printing and mailing.

Altair Users Group

All *Altair 680* purchasers will receive a free one year membership to the Altair Users Group. This group is the largest of its kind in the world and includes thousands of *Altair 8800* and *680* users.

Members of the Altair Users Group are kept abreast of Altair developments through the monthly publication, ***Computer Notes.***

Altair 680 Documentation

The *Altair 680* kit comes with complete documentation including assembly manual, assembly hints manual, operation manual, and theory manual. Assembled units come with operation and theory manuals. Turnkey mode! and CPU boards also include documentation.

NOTE: *A complete set of Altair 680 manuals in a 3-ring Altair binder is now available for $14.50 (regularly $25). Offer expires January 30, 1976.*

Delivery

Personal checks take 2-3 weeks to process while money orders and credit card purchases can be processed in 1-3 days. Delivery should be 30-60 days but this can vary according to order backlog. All orders are handled on a first come, first served basis.

Altair 680 Prices

***Altair 680* complete computer kit**	**$345**
***Altair 680* assembled and tested**	**$420**
***Altair 680T* turnkey model (complete *Altair 680* except front panel control board) Kit Only**	**$280**
***Altair 680* CPU board (including pc board, 6800 microprocessor chip, 1024 word memory, 3 way interface and all remaining components except power supply)**	**$195**
***Altair 680* CPU board assembled and tested**	**$275**
Option IC socket kit (contains 40 IC sockets. CPU, memory and PROM sockets come with 680 kit)	**$ 29**
Option cooling fan (required when expanding 680 internally)	**$ 22**
Option cooling fan installed	**$ 26**
PROM kit (256 x 8-bit ultraviolet, erasable 1702 devices)	**$ 25**
Connectors (Two sets of 25-pin connectors. Required when interfacing 680 to external devices)	**$ 22**

Prices, delivery and specifications subject to change.

MITS
"Creative Electronics"

MITS/6328 Linn N.E., Albuquerque, NM 87108 505/262-1951

MAIL THIS COUPON TODAY!

☐ Enclosed is check for $________________.
☐ BankAmericard #____________ ☐ or Master Charge #__________
☐ Altair 680 ☐ Kit ☐ Assembled ☐ Altair 680T ☐ Altair 680CPU
☐ Socket Kit ☐ Fan ☐ PROM kit ☐ Documentation Special
☐ Please send literature
NAME________________________________
ADDRESS______________________________
CITY____________________ STATE & ZIP__________
MITS/6328 Linn N.E., Albuquerque, NM 87108 505/262-1951